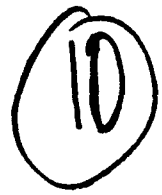
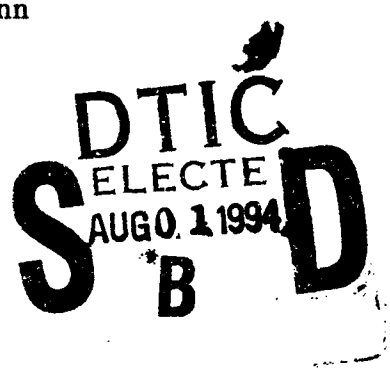


Geometric Sensing of Known Planar Shapes

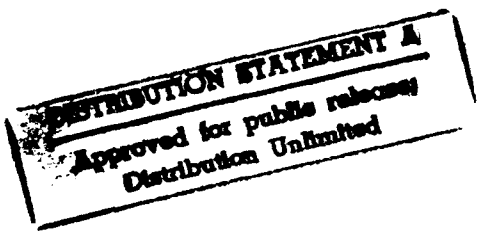
Yan-Bin Jia and Michael Erdmann

CMU-RI-TR-94-24



The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

July 1994



Support for this research was provided in part by Carnegie Mellon University, and in part by the National Science Foundation through the following grants: NSF Research Initiation Award IRI-9010686, NSF Presidential Young Investigator award IRI-9157643, and NSF Grant IRI-9213993.

Carnegie Mellon

The Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213-3890

22 July 1994

Defense Technical Information Center
Cameron Station
Alexandria, VA 22314

RE: Report No. CMU-RI-TR-94-24

Permission is granted to the Defense Technical Information Center and the National Technical Information Service to reproduce and sell the following report, which contains information general in nature:

Yan-Bin Jia and Michael Erdmann
Geometric Sensing of Known Planar Shapes

Yours truly,

Marcella L. Zaragoza

Marcella L. Zaragoza
Graduate Program Coordinator

enc.: 12 copies of report

Contents

List of Figures

List of Tables

Abstract

Industrial assembly involves sensing the pose (orientation and position) of a part. Efficient and reliable sensing strategies can be developed for an assembly task if the shape of the part is known in advance. In this paper we investigate two problems of determining the pose of a polygonal part of known shape. In the first problem, the part has a continuum of possible poses, while in the second problem, it has a finite number of possible poses.

More specifically, the first problem involves determining the pose of a convex n-gon from a set of m supporting cones, that is, cones with both sides supporting the polygon. An algorithm with running time $O(nm)$ which almost always reduces to $O(n+m\log n)$ is presented to solve for all possible poses of the polygon. As a consequence, the *polygon inscription* problem of finding all possible poses for a convex n-gon inscribed in another convex m-gon, can be solved within the same asymptotic time bound. We prove that the number of possible poses cannot exceed $6n$, given $m \geq 2$ supporting cones with distinct vertices. Experiments demonstrate that two supporting cones are sufficient to determine the real pose of the n-gon in most cases. Our results imply that sensing in practice can be carried out by obtaining viewing angles of a planar part at multiple exterior sites in the plane. As a conclusion, we generalize this and other sensing methods into a scheme named *sensing by inscription.*

On many occasions a parts feeder will have reduced the number of possible poses of a part to a small finite set. In order to distinguish between the remaining poses of the part some simple sensing or probing operation may be used. Our second problem, called *sensing by point sampling*, is concerned with finding the minimum number of sensing points required to distinguish between a finite set of polygonal shapes. In practice this can be carried out by embedding a series of point light detectors in a feeder tray or by using a set of mechanical probes that touch the feeder at a finite number of predetermined points.

Intuitively, each sensing point can be regarded as a binary bit that has two values 'contained' and 'not contained'. So the robot senses a shape by reading out the binary representation of the shape, that is, by checking which points are contained in the shape and which are not. The formalized sensing problem: Given n polygons with a total of m edges in the plane, locate the fewest points such that each polygon contains a distinct subset of points in its interior. We show that this problem is equivalent to an NP-complete set-theoretic problem introduced as *Discriminating Set.* By a reduction to Hitting Set (and hence to Set Covering), an $O(n^2m^2)$ approximation algorithm is presented to solve the sensing problem with a ratio of $2\ln n$. Based on a reverse reduction, we prove that one can use an algorithm for Discriminating Set with ratio $c\log n$ to construct an algorithm for Set Covering with ratio $c\log n+O(\log\log n)$. Thus approximating Discriminating Set exhibits the same hardness as that of approximating Set Covering recently shown in [44] and [7]; this result implies that the ratio $2\ln n$ is asymptotically optimal unless NP $\subset$ DTIME($n^{\text{poly}\log n}$). We also analyze the complexity of subproblems of Discriminating Set, based on their relationship to a generalization of Independent Set called *3-Independent Set.* Finally, we present experimental results showing that sensing by point sampling is mostly applicable when poses are densely distributed in the plane.

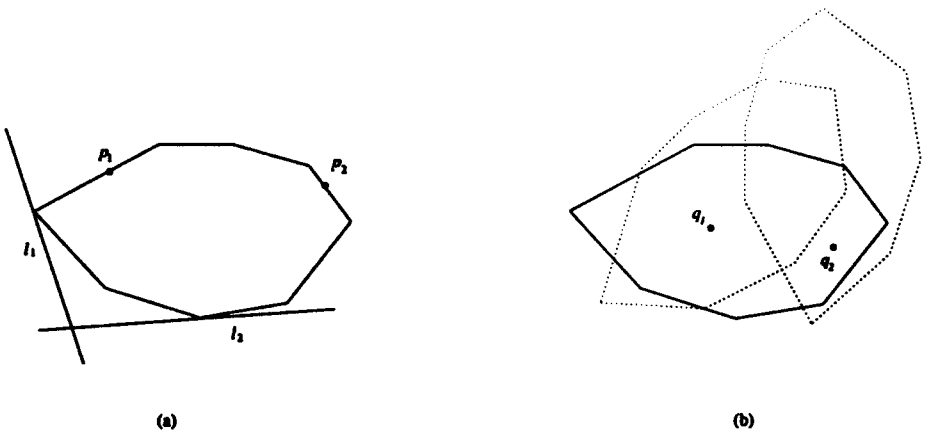


Figure 1: Geometric sensing of an 8-gon. (a) The case of a continuum of possible poses: Four incidence constraints p_1, p_2, l_1 and l_2 suffice to determine the pose. (b) The case of a finite number (three) of possible poses: Only the containment of two points q_1 and q_2 needs to be checked to determine the real pose.

1 Introduction

Sensing a part involves determining both its shape and pose. By pose we mean the position as well as the orientation of the part. Prior to selecting a sensing method, we often will make some assumptions about the shape of the part to be sensed. The resulting sensing method is affected greatly by what is known about the shape. For instance, without making any assumptions, we might not even be able to start segmentation of the part image, whereas knowing that the shape is convex polygonal, we can employ some simple non-vision technique such as finger probing. An effective sensing method should make use of its knowledge about the part shape *as much as possible* to attain simplicity, efficiency and robustness.

Parts in many assembly applications are manufactured to high precisions, so we can make the assumption that their shapes are known reasonably well in advance. Accordingly, the design of sensing strategies should be based on the geometry of parts. The task of sensing reduces to obtaining enough geometric constraints which, when combined with the part geometry, suffice to derive the part poses. Consequently *minimizing* the necessary geometric constraints becomes very important for reducing the sensing complexity. In this paper, we propose two approaches for sensing polygonal parts in known shapes, one applicable to a continuum of possible poses and the other applicable to finite possible poses.

Perhaps the simplest geometric constraint on a polygon is *incidence*—when some edge touches a fixed point or some vertex is on a fixed line. For instance, Figure 1(a) shows an 8-gon constrained by two points p_1, p_2 and two lines l_1, l_2. The question we want to ask is: Generally how many such constraints are necessary to *fix* the polygon in its real position. Note that any two such incidence constraints will confine all possible positions to a *locus curve* which consists of a finite number of algebraic curves parameterized by the part's orientation. Three constraints, as long as not defined by collinear points or concurrent lines, will allow only a finite number of valid poses. These poses occur when different locus curves, each given by a pair of constraints, intersect at the same orientations. An upper bound on

the number of possible poses can be analyzed. The addition of a fourth constraint is usually enough to reduce this number to one—the real pose.

If all incidences are given by lines, sensing can be viewed as inscribing the polygon into a larger polygon (not necessarily bounded) defined by these lines; if all constraints are given by points, sensing can be viewed as placing the polygon defined by these points into the sensed polygon.

Point constraints can be created by "probing" the polygon along various directions with a tactile sensor or a range finder. Line constraints can be obtained with an angular sensor scanning across the object at exterior sites. In Section 2 we study the case of sensing with line constraints only, offering a very efficient algorithm to solve for all possible poses. We give a tight upper bound on the number of poses for three lines, and conduct experiments to show that four lines are practically enough to determine the real pose.

The set of possible poses can often be reduced from a continuum to a finite number in advance, by planned manipulations such as pushing or squeeze-grasping, or by sensing from geometric constraints as mentioned above. More specialized sensing methods can be devised to distinguish between the remaining finite number of poses. We now view each pose as a closed set of points in the plane occupied by the part in that pose, so that sensing becomes distinguishing point sets, say $P_1, \ldots, P_n$. An easy way is to sample several points, checking which of them are contained by the current point set (pose). Suppose the same 8-gon is known to have only three possible poses, as shown in Figure 1(b), then the real pose (drawn in solid line) can be determined by verifying that it contains both points q_1 and q_2. This can be implemented in a number of different ways, for instance, by placing light detectors underneath the point locations or by probing the point locations from above with a robot finger.

In Section 3 we address the problem of distinguishing finite polygons by point sampling. We prove that minimizing the total number of sampling points is NP-complete, and offer an approximation algorithm with a greedy heuristics. The algorithm produces a set of sampling points whose size is within a factor of $2 \log n$ of the optimal. We also exhibit a proof demonstrating the hardness of improving this approximation ratio.

2 Sensing by Inscription

2.1 The Inscription Problem

In this section we will study the problem of detecting the pose of a convex polygon in the plane by taking views of the polygon from multiple exterior sites. The shape of the polygon is assumed to be *known* in advance, but the pose of the polygon can be arbitrary. Each view results in a *cone* formed by the two outermost occluding rays starting from the viewing site. The cone in turn imposes a constraint on the possible poses of the polygon—the polygon must be contained in the cone and make contact with both its sides. A containment in which every edge of the containing object touches the contained object is called an *inscription*, so we shall say that the polygon is *inscribed* in the cone. Such constraints imposed by individual views together allow only a small number of possible poses of the polygon, which often reduces to one. For example, Figure 2 illustrates two views taken of a convex 6-gon P in

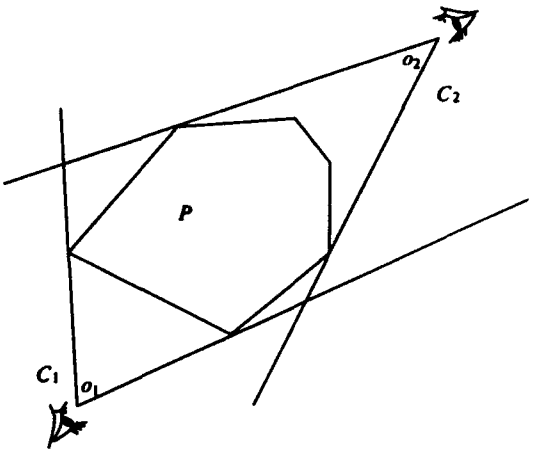


Figure 2: Sensing the pose of a polygon by taking two views.

some unknown pose from sites o_1 and o_2 respectively: The two cones C_1 and C_2 thus formed determine the real pose of P, and this pose can be solved using the algorithm presented later in Section 2.2.4.

The above sensing approach appears to be simple, but to make it efficient and to minimize the cost of sensing hardware we would like to take as few views as possible. This leads us to the main question of this section: *How many views are sufficient in the general case in order to determine the pose of a convex n-gon?*[1]

The answer to the above question is *two*, and to argue this answer we will go through several steps each of which occupies a separate subsection: Section 2.2 describes how to compute the set of possible poses for a convex polygon inscribed in multiple cones, and derives an upper bound on the number of possible poses for two-cone inscription in particular; Section 2.3 empirically demonstrates that, in spite of the upper bound, two cones turn out to be sufficient in most cases to uniquely determine the pose of an inscribed polygon. Section 4 further discusses the extensions of this method and proposes a general sensing scheme—sensing by inscription.

2.1.1 Related Work

Canny and Goldberg [11] have introduced a Reduced Intricacy in Sensing and Control (RISC) paradigm that aims at improving the accuracy, speed and robustness of sensing by coupling simple and specialized hardware with fast and task-oriented geometric algorithms.

The *cross-beam sensing* method developed in [55] finds the orientation of a polygon (or polyhedron) by measuring its diameters along three different directions and comparing the measurements with the precomputed diameter function [27]; then it solves a vertex-line correspondence problem for the position of the polygon by least squares fitting. This

[1]It should be noted that there exist cases in which the pose of a convex n-gon cannot be uniquely determined no matter how many views are taken. This happens only if the polygon preserves self-congruence over certain rotations. (It is not hard to see that in such a case the rotation angle must be a multiple of $\frac{2\pi}{k}$, where k is a positive integer such that $k|n$.) However, all congruent poses are usually considered as the same in the real applications.

method essentially determines the pose by inscribing the polygon in a hexagon constructed from the sensory data. The idea of characterizing shapes with diameters and chords was also addressed earlier in [53].

For the special case that the poses are finite, [49] presents an efficient method of placing a *registration mark* on the object so that the pose can be recognized by locating the mark position (with a simple vision system). For robustness to sensor imperfections, the marked point maximizes the distance between the nearest pair among its possible locations. For the case that the number of parts is finite, [28] recognizes a part with a modified parallel-jaw gripper by a sequence of grasp actions and diameter measurements.

Model-based recognition and localization can often be regarded as a *constraint satisfaction* problem which searches for a consistent matching between sensory data (e.g., 2-D) and model(s) (e.g., 3-D) based on the geometric constraints between them. (See [30].) [26] discusses how to identify and locate a polyhedron on a known plane using local information from tactile sensors which includes the position of contact points and the ranges of surface normals at these points. Motivated by an interpretation tree developed in [26], [52] determines the pose of an object grasped by a hand, under a situation very close to inscription. Also using an interpretation tree search, [37] solves the pose of a polyhedra by matching a set of 3-D line segments, obtained by three light-stripe range finders, to model faces; then the pose uncertainty is estimated using the covariance matrix of the endpoints of these line segments. [14] uses a polynomial approach to solve for the line-to-plane correspondences involved in pose determination. Based on a generalized Hough transform, [42] estimates the pose of a 3-D object by matching point triples on the object to possibly corresponding triples in the image.

In the meantime, a variety of polygon shape descriptors [2], [45] have been analyzed theoretically and/or demonstrated experimentally to be efficient and robust to uncertainties.

Geometric algorithms for sensing unknown poses as well as unknown shapes have also been studied. Cole and Yap [16] considered "finger" probing a convex n-gon (n unknown) along directed lines, and gave a procedure guaranteed to determine the n-gon with $3n$ probes. This work was later extended in [18] which investigates the complexity of various models for probing convex polytopes in d-dimensional Euclidean space. Using a more powerful probe model that returns not only the contact points but also the normals at these points, Boissonnat and Yvinec [9] showed how to compute the exact shape of a simple polygon as long as some mild conditions about the shape are met. [40] gives algorithms of projection probing and line probing (similar to inscription) which perform the numbers of probes that are both sufficient and necessary to determine a convex n-gon. Close upper and lower bounds are derived in [41] on the number of composite probes to reconstruct a convex n-gon, where a composite probe comprises in parallel several supporting line probes or finger probes.

The polygon containment problem, that is, deciding whether an n-gon P can fit into an m-gon Q under translations and/or rotations, has been studied by various researchers in computational geometry. (See [5], [6], [12], [23].) In the case where Q is convex, the best known algorithm runs in time $O(m^2n)$ when both translations and rotations are allowed [12]. Here we will deal with a special case of containment in which each edge of Q must touch P; this constraint causes a reduction of the running time to $O(mn)$, or $O(n + m \log n)$ in practical situations.

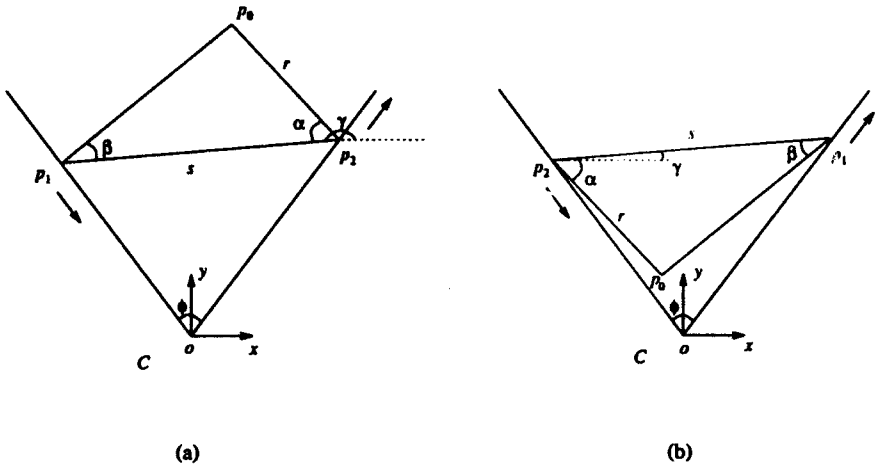


Figure 3: A triangle sliding in a cone. Vertices p_1 and p_2 move along the two sides of cone C. The locus of p_0 is an elliptic curve (possibly degenerating into a line segment) parameterized by the angle γ between directed edge $\overrightarrow{p_2p_1}$ and the x axis. There are two different cases: (a) p_0 is above edge p_1p_2; (b) p_0 is below edge p_1p_2.

2.2 Multi-Cone Inscription

To simplify the presentation, let us *agree* throughout Section 2.2 that all angles take values in the half-open interval $[0, 2\pi)$. In accordance with this agreement, any expression E on angles equated with or assigned to an angle θ in formulas such as $\theta = E$ or $\theta \leftarrow E$ will be regarded as 'E mod 2π', even though we do not mention so explicitly. (However, this does not apply to conditions such as $E_1 = E_2$ where both sides are expressions on angles.) Moreover, intervals for angle values whose left endpoints are greater than right endpoints are allowed; for example, an interval $[\alpha, \beta]$, where $0 \leq \beta < \alpha < 2\pi$, is understood as the interval union $[\alpha, 2\pi) \cup [0, \beta]$.

2.2.1 A Triangle Sliding in a Cone

We first deal with the case of a triangle in a cone, not only because it is the simplest but also because the case of a polygon, as we will see later, can be decomposed into subcases of triangles. Let $\triangle p_0p_1p_2$ be a triangle inscribed in an upright cone C with angle ϕ and vertex o, where $0 < \phi < \pi$, making contacts with both sides of the cone at vertices p_1 and p_2 respectively. What is the locus of vertex p_0 as edge p_1p_2 slides against the two sides of the cone?

Two different situations can occur with this inscription: (a) p_0 is outside $\triangle p_1op_2$, and (b) p_0 is inside (only when $\angle p_1p_0p_2 \geq \phi$). (See Figure 3.) Assume that $\triangle p_0p_1p_2$ may degenerate into any one of its edges but not a point; writing $\alpha = \angle p_0p_2p_1$, $\beta = \angle p_0p_1p_2$, $r = |p_0p_2|$ and $s = |p_1p_2|$, this degeneracy is taken into account by the following constraints:

$$0 \leq \alpha, \beta < \pi, \quad s \geq 0, \quad \text{and} \quad \begin{cases} r > 0, & \text{if } s = 0 \text{ or } \alpha > 0; \\ 0 \leq r \leq s, & \text{otherwise.} \end{cases}$$

Let us set up a coordinate system with the origin at o and the y axis bisecting angle ϕ, as

shown in Figure 3. Then the orientation of $\triangle p_0p_1p_2$ can be denoted by the angle γ between vector $\overrightarrow{p_2p_1}$ and the x axis. The range of valid γ values can be easily determined as

$$\max(\frac{\pi}{2}+\frac{\phi}{2}, \frac{\pi}{2}-\frac{\phi}{2}+\alpha) \leq \gamma \leq \min(\frac{3\pi}{2}-\frac{\phi}{2}, \frac{3\pi}{2}+\frac{\phi}{2}-\beta)$$

in case (a), and as $\gamma = \gamma' \bmod 2\pi$ in case (b), where

$$\frac{3\pi}{2}+\frac{\phi}{2}+\alpha \;\leq\; \gamma' \;\leq\; 2\pi+\frac{\pi}{2}-\frac{\phi}{2}-\beta.$$

For any valid γ, there exists a unique pose of the triangle in cone C; this allows us to parameterize the locus (x,y) of p_0 by γ.

The two cases (a) and (b) yield results that resemble too much to each other to be treated separately here. We begin with writing out the following equations for the locus of p_0:

$$\begin{aligned} x &= r\cos(\gamma-\alpha) \pm |op_2| \sin\frac{\phi}{2}; \\ y &= r\sin(\gamma-\alpha) + |op_2| \cos\frac{\phi}{2}, \end{aligned}$$

where $\frac{|op_2|}{s} = \frac{\mp\cos(\gamma\mp\frac{\phi}{2})}{\sin\phi}$. Here the notation '$\pm$' means '$+$' in case (a) and '$-$' in case (b) and the notation '$\mp$' means just the opposite. Several steps of manipulation on the above equations plus a detailed subsequent analysis on the ranges of angles reveal the locus of p_0 as described below.

Namely, as edge p_1p_2 slides in the cone, p_0 traces out an *elliptic* curve $\mathcal{C}$ with implicit equation

$$ax^2 \pm bxy + cy^2 = d,$$

where

$$\begin{aligned} a &= r^2 - rs\frac{\sin(\frac{\phi}{2}\mp\alpha)}{\sin\frac{\phi}{2}} + \frac{s^2}{2-2\cos\phi}; \\ b &= \frac{(2r\cos\alpha - s)s}{\sin\phi}; \\ c &= r^2 - rs\frac{\cos(\frac{\phi}{2}\mp\alpha)}{\cos\frac{\phi}{2}} + \frac{s^2}{2+2\cos\phi}; \\ d &= \Big(r(r - s\frac{\sin(\phi\mp\alpha)}{\sin\phi})\Big)^2. \end{aligned}$$

Furthermore, if the orientation γ changes monotonically within its range, p_0 moves *monotonically* along $\mathcal{C}$ except when $\mathcal{C}$ degenerates into a line segment. In that degenerate case, p_0 moves along a segment of a line through the the cone vertex o and with equation

$$\begin{cases} \cos\frac{\phi}{2}\,x \;\mp\; \sin\frac{\phi}{2}\,y = 0, & \text{if } r = 0 \text{ (and thus } \alpha = 0); \\ \cos(\frac{\phi}{2}\mp\alpha)\,x \;\pm\; \sin(\frac{\phi}{2}\mp\alpha)\,y = 0, & \text{if } r \neq 0,\ s \neq 0, \text{ and } \frac{r}{s} = \frac{\sin(\phi\mp\alpha)}{\sin\phi}, \end{cases}$$

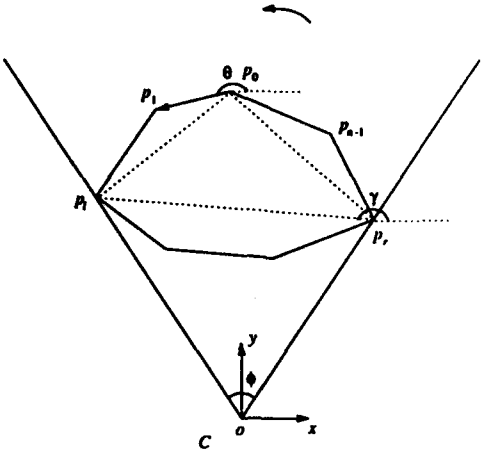


Figure 4: A convex polygon P rotating in a cone. The pose of P is denoted by the position of vertex p_0 and the angle θ between directed edge $\overrightarrow{p_0p_1}$ and the x axis. The space of orientations $[0, 2\pi)$ is partitioned into closed intervals, each defining an elliptic curve that describes the corresponding locus of p_0.

crossing the same point at any two valid orientations $\gamma_1 \neq \gamma_2$ with $\gamma_1 + \gamma_2 = 2\pi \pm \phi$ when the line segment assumes the first equation, or with $\gamma_1 + \gamma_2 = 2\pi - \phi + 2\alpha$ in case (a) and with $\gamma_1 + \gamma_2 = 2\alpha + \phi$ in case (b) when it assumes the second equation.

Interestingly, we observe that in case (a) if $r \neq 0$ and $\frac{r}{s} = \frac{\sin(\phi-\alpha)}{\sin\phi}$, the second line equation reduces to $x = 0$ when $\phi = 2\alpha$ in case (a); but this reduction does not occur in case (b) because $\frac{r}{s} = \frac{\sin(\phi+\alpha)}{\sin\phi}$ implies that $0 < \phi + 2\alpha < 2\pi$, thereby establishing $\sin(\frac{\phi}{2} + \alpha) > 0$. This observation reflects a small asymmetry between the two cases.

2.2.2 One-Cone Inscription

Now consider the case that a convex n-gon P with vertices $p_0, p_1, \ldots, p_{n-1}$ in counterclockwise order is inscribed in a cone C. Let us choose the same coordinate system as in Section 2.2.1. Then the pose of P is uniquely denoted by the locus of some vertex, say p_0, and the angle θ between the x axis and some directed edge, say $\overrightarrow{p_0p_1}$. Clearly any orientation θ gives rise to a unique pose of P; so we can compute the locus of p_0 as a function of θ over $[0, 2\pi)$. Let p_l and p_r be the two vertices currently incident on the left and right sides of cone C respectively. (See Figure 4.) As long as p_l and p_r remain incident on these two sides respectively, the problem reduces to the case of $\triangle p_0p_lp_r$ sliding in cone C except that the locus of p_0 (an elliptic curve) now needs to be parameterized by θ, instead of γ which we had used before. This is easy for we observe that, when $p_l \neq p_r \neq p_0$,

$$\gamma = \begin{cases} \theta + \angle p_1p_0p_r + \angle p_0p_rp_l - \pi, & \text{if } p_0 \text{ is above edge } p_lp_r; \\ \theta + \angle p_1p_0p_l + \angle p_0p_lp_r - \pi, & \text{otherwise.} \end{cases}$$

The three special cases $p_l = p_0$, $p_r = p_0$ and $p_l = p_r$ can be handled by substituting π for $\angle p_1p_0p_r$ and $\angle p_1p_0p_r$ for $\angle p_0p_rp_l$ for the first case, π for $\angle p_1p_0p_l$ and $\angle p_1p_0p_l$ for $\angle p_0p_lp_r$ for the second case, and 0 for $\angle p_0p_rp_l$ (or $\angle p_0p_lp_r$) for the third case.

More observations show that the entire range $[0, 2\pi)$ of orientations can be partitioned into a sequence of closed intervals, within each of which the vertices p_l and p_r incident on cone C are invariant.

We present a linear-time algorithm that computes the above orientation intervals as well as the corresponding elliptic curves describing the locus of p_0. The algorithm rotates the polygon counterclockwise in the cone, generating a new interval whenever one (or both) of the incident vertices p_l and p_r changes; the new incident vertex (or vertices) is determined by a comparison between angle ϕ and the angle intersected by the two rays $p_{l-1}p_l$ and $p_r p_{r-1}$.

Let $[\theta_{\min}, \theta_{\max}]$ denote the current interval, and let φ_i denote the interior angle $\angle p_{i-1}p_i p_{i+1}$ for $0 \leq i \leq n-1$. (For convenience, arithmetic operations performed on the subscripts of vertices or internal angles are regarded as followed by a 'mod n' operation; for example, p_{-1} is identified with p_{n-1} and p_n with p_0.) In the algorithm, Φ_{left} and Φ_{right} keep track of the angle between $\overrightarrow{p_l p_{l+1}}$ and $\overrightarrow{p_0 p_1}$ and the angle between $\overrightarrow{p_r p_{r+1}}$ and $\overrightarrow{p_0 p_1}$ respectively. The algorithm proceeds as follows:

Algorithm 1

Step 1 Start at the pose such that edge p_0p_1 aligns with the right side of C. Locate the vertex p_i in contact with the left side of C. Set $\Phi_{\text{left}} \leftarrow \sum_{j=l+1}^{n}(\pi - \varphi_j)$, $\Phi_{\text{right}} \leftarrow 0$, $\theta_{\min} \leftarrow \frac{\pi}{2} - \frac{\phi}{2}$, $l \leftarrow i$, and $r \leftarrow 0$.

Step 2 The current orientation interval has its left endpoint at $\theta_{\min}$. Output the elliptic curve (now parameterized by θ) resulting from sliding edge $p_l p_r$ in cone C.

Next determine the right endpoint $\theta_{\max}$ of the current interval. Let ψ be the angle intersected by rays $p_{l-1}p_l$ and $p_r p_{r-1}$; set $\psi \leftarrow nil$ if they do not intersect. There are three different cases:

Case 1 $\psi < \phi$ or $\psi = nil$. [Advance p_r clockwise to the next vertex.] Set $\Phi_{\text{right}} \leftarrow \Phi_{\text{right}} + \pi - \varphi_r$, $\theta_{\max} \leftarrow \Phi_{\text{right}} + \frac{\pi}{2} - \frac{\phi}{2}$, and $r \leftarrow r - 1$.

Case 2 $\psi > \phi$. [Advance p_l.] Set $\Phi_{\text{left}} \leftarrow \Phi_{\text{left}} + \pi - \varphi_l$, $\theta_{\max} \leftarrow \Phi_{\text{left}} + \frac{3\pi}{2} + \frac{\phi}{2}$, and $l \leftarrow l - 1$.

Case 3 $\psi = \phi$. [Advance both p_l and p_r.] Set $\Phi_{\text{left}} \leftarrow \Phi_{\text{left}} + \pi - \varphi_l$, $\Phi_{\text{right}} \leftarrow \Phi_{\text{right}} + \pi - \varphi_r$, $\theta_{\max} \leftarrow \Phi_{\text{left}} + \frac{3\pi}{2} + \frac{\phi}{2}$, $l \leftarrow l - 1$, and $r \leftarrow r - 1$.

Output the current interval $[\theta_{\min}, \theta_{\max}]$. Set $\theta_{\min} \leftarrow \theta_{\max}$ and repeat Step 2 until r changes from 1 to 0.

The number of intervals produced by the above algorithm cannot exceed $2n$, because each loop of Step 2 advances either p_r to p_{r-1}, or p_l to p_{l-1}, or both to p_{r-1} and p_{l-1} respectively, and because there are $2n$ vertices in total (n each for p_l and p_r) to advance before returning to the initial incident vertices p_0 and p_i.

We can easily apply the above algorithm for the general case in which the vertex of cone C is at an arbitrary point (x_0, y_0) and the axis of the cone forms an angle θ_0 with the y axis. Each generated interval $[\alpha, \beta]$ now needs to be right shifted to $[\alpha + \theta_0, \beta + \theta_0]$. Let $q = p_r$ and

$q' = p_l$ if p_0 is above edge $p_l p_r$, and let $q = p_l$ and $q' = p_r$ otherwise. Then the corresponding locus (x, y) of p_0 is determined as, assuming $p_l \neq p_r \neq p_0$,

$$\begin{aligned} x &= \left(-|qp_0| \cdot \cos \angle p_1 p_0 q + |p_l p_r| \cdot k_c \cdot \frac{\sin(\frac{\phi}{2} \mp \theta_0)}{\sin \phi}\right) \cos\theta \\ &\quad + \left(|qp_0| \cdot \sin \angle p_1 p_0 q - |p_l p_r| \cdot k_s \cdot \frac{\sin(\frac{\phi}{2} \mp \theta_0)}{\sin \phi}\right) \sin\theta + x_0; \\ y &= \left(-|qp_0| \cdot \sin \angle p_1 p_0 q \pm |p_l p_r| \cdot k_c \cdot \frac{\cos(\frac{\phi}{2} \mp \theta_0)}{\sin \phi}\right) \cos\theta \\ &\quad - \left(|qp_0| \cdot \cos \angle p_1 p_0 q \pm |p_l p_r| \cdot k_s \cdot \frac{\cos(\frac{\phi}{2} \mp \theta_0)}{\sin \phi}\right) \sin\theta + y_0, \end{aligned}$$

where $k_c = \cos(\angle p_1 p_0 q + \angle p_0 q q' - \theta_0 \mp \frac{\phi}{2})$ and $k_s = \sin(\angle p_1 p_0 q + \angle p_0 q q' - \theta_0 \mp \frac{\phi}{2})$. Here '$\pm$' and '$\mp$' denote '+' and '−', or '−' and '+', according as p_0 is above or below edge $p_l p_r$.

2.2.3 Upper Bounds

The preceding subsection tells us that the set of possible poses for a convex polygon inscribed in one cone can be described by a continuous and piecewise elliptic curve defined over orientation space $[0, 2\pi)$. We call this curve the *locus curve* for the inscription. This subsection will show that only finite possible poses exist for a convex n-gon P inscribed in two cones, so long as the vertices of the cones do not coincide. An upper bound on the number of possible poses can be obtained straightforwardly by intersecting two locus curves, each resulting from the inscription of P in one cone.

Claim 1 *There exist no more than $8n$ possible poses for a convex n-gon P inscribed in two cones C_1 and C_2 with distinct vertices.*

Proof. Let $p_0, \ldots, p_{n-1}$ be the vertices of P in counterclockwise order; then a pose of P can be represented by the location of p_0 as well as the angle θ between directed edge $\overrightarrow{p_0 p_1}$ and the x axis. Let $C_1(\theta)$ and $C_2(\theta)$ be the two locus curves, for the inscriptions of P in cones C_1 and C_2 respectively. We only need to show that C_1 and C_2 meet at most $8n$ times, that is, they pass through common points at no more than $8n$ values of θ.

It is known that each C_i consists of at most $2n$ elliptic curves defined over a sequence of intervals that partition $[0, 2\pi)$. Intersecting these two sequences of intervals gives a partition that consists of at most $4n$ intervals. Within each interval the possible orientations (hence the possible poses) of P can be found by computing where the corresponding pair of elliptic curves meet. According to the last subsection, this pair of curves may be written in the parameterized forms

$$(a_{ix} \cos\theta + b_{ix} \sin\theta + x_i,\ a_{iy} \cos\theta + b_{iy} \sin\theta + y_i),$$

for $i = 1, 2$. Here (x_i, y_i) is the vertex of cone C_i, and $a_{ix}, b_{ix}, a_{iy}, b_{iy}$ are constants determined by P and C_i. Using the condition $(x_1, y_1) \neq (x_2, y_2)$, we suppose $x_1 \neq x_2$ without loss of

generality, and let $\Delta = \sqrt{(a_{1x} - a_{2x})^2 + (b_{1x} - b_{2x})^2}$. Then it is not hard to show that these two curves do not meet if $|x_1 - x_2| > \Delta$. Otherwise they may meet only at

$$\theta = \beta - \alpha \qquad \text{and} \qquad \theta = \pi - \beta - \alpha,$$

where $\alpha = \text{atan}(\frac{a_{1x}-a_{2x}}{\Delta}, \frac{b_{1x}-b_{2x}}{\Delta})$ and $\beta = \sin^{-1}\frac{x_2-x_1}{\Delta}$. □

The upper bound $8n$ is *not* tight: A lower one can be obtained even without using two cones to constrain the polygon. Notice in the proof above that the bound came from a partition of orientation space $[0, 2\pi)$ into up to $4n$ intervals which combined the individual pose constraints imposed by the two cones. Therefore an improvement on that bound must require a different partitioning of $[0, 2\pi)$. To see this, we regard each cone as the intersection of two half-planes and decompose its constraint on the polygon into two constraints introduced by the half-planes independently.

A polygon P is said to be *embedded* in a half-plane h if P is contained in h and supported by its bounding line. Thus P is inscribed in a cone if and only if it is embedded in the two half-planes defining the cone by intersection. Two cones with distinct vertices together provide three or four half-planes, of which no three have *concurrent* bounding lines, i.e., bounding lines that pass through a common point. Such three half-planes are indeed enough to bound the number of possible poses of P within $6n$.

Theorem 1 *There exist no more than 6n possible poses for a convex n-gon P embedded in three half-planes with non-concurrent bounding lines; furthermore, this upper bound is tight.*

Proof. Let l_1, l_2 and l_3 be the bounding lines of the three half planes respectively. We can assume that these lines are not all parallel; otherwise it is easy to see that no feasible pose for P exists. So suppose l_1 and l_2 intersect; their corresponding half-planes form a cone in which P is inscribed. Let the orientation of P be represented by the angle θ between the x axis and some directed edge of P. Then orientation space $[0, 2\pi)$ is partitioned into at most $2n$ intervals, according to which pair of vertices are possibly on l_1 and l_2 respectively. In the mean time, it is also partitioned into exactly n intervals, according to which vertex is possibly on l_3. Intersecting the intervals in these two partitions yields a finer partition of $[0, 2\pi)$ that consists of at most $3n$ intervals, each containing orientations at which P is to be supported by l_1, l_2 and l_3 at the same three vertices.

Let us look at one such interval, and let p_i, p_j and p_k be the vertices of P on l_1, l_2 and l_3 respectively whenever a possible orientation exists in the interval. The possible orientations occur exactly where l_3 crosses an elliptic curve $\mathcal{C}(\theta)$ traced out by p_k when sliding p_i and p_j on l_1 and l_2 respectively. Now we prove that this interval contains at most two possible orientations. Note $\mathcal{C}$ does not degenerate into a point because the case $p_i = p_j = p_k$ will never happen, given l_1, l_2 and l_3 are not concurrent. Therefore $\mathcal{C}$ is either an elliptic segment monotonic in θ or a line segment that attains any point for at most two θ values. (See Section 2.2.1.) In both cases, it is clear that $\mathcal{C}$ crosses l_3 for no more than two θ values. Thus there are at most $6n$ possible poses in orientation space $[0, 2\pi)$.

Appendix A gives an example in which a polygon can actually have $6n$ poses when embedded in three given half-planes, thereby proving the tightness of this upper bound. □

When two of the three non-concurrent bounding lines are parallel, we can similarly derive a smaller but tight upper bound $4n$, using the same interval partitioning technique. We omit the details of the derivation but give a worst-case example in Appendix A.

Since any two cones with distinct vertices are formed by three or four half-planes with non-concurrent bounding lines, and since embedding a polygon in three half-planes with non-concurrent bounding lines is equivalent to inscribing it in any two cones determined by intersecting a pair of the half-planes, we immediately have

Corollary 1 *There exist at most $6n$ possible poses for a convex n-gon inscribed in two cones with distinct vertices, and this upper bound is tight.*

Would more half-planes (or cones) further reduce the number of possible poses for an embedded polygon to be asymptotically less than n? The answer is no. For example, an embedded regular n-gon will always have kn possible poses, where $1 \leq k \leq 6$, no matter how many half-planes are present. The experimental results in Section 2.3 will show that two cones (or four half-planes) are usually sufficient to determine a unique pose for the polygon.

2.2.4 An Algorithm for Inscription

With the results in the previous subsections, we here present an algorithm that computes all possible poses for a convex n-gon P to be inscribed in m cones $C_1, \ldots, C_m$, where $m \geq 2$. (The vertices of these cones are assumed to be distinct.) Let $p_0, \ldots, p_{n-1}$ be the vertices of P in counterclockwise order.

Algorithm 2

Step 1 [Compute an initial set of poses w.r.t. two cones.] Solve for all possible poses of P when inscribed in cones C_1 and C_2 (use Algorithm 1 and see the proof of Claim 1), and let set S consist of the resulting poses (already sorted by orientation). Set $i \leftarrow 3$.

Step 2 [Verify with the remaining cones.] If $i = m + 1$ or $S = \emptyset$ then terminate. Otherwise go to Step 3 if $|S| = 1$ or $|S| = 2$. Otherwise apply Algorithm 1 to generate the locus curve $C_i(\theta)$ for the inscription of P in C_i. Sequentially verify whether each pose in S is on $C_i(\theta)$, deleting from S those poses that are not. Set $i \leftarrow i + 1$ and repeat Step 2.

Step 3 [More efficiently verify one or two poses.] For each pose in S, let polygon P' be P in that pose and do the following: For $i \leq j \leq m$ construct the supporting cone C'_j of P' at the vertex of cone C_j; if there exists some C'_j that does not coincide with C_j, then delete the corresponding pose from S.

When the above algorithm terminates, set S will contain all possible poses for the inscription. Corollary 1 shows that there are at most $6n$ poses in S after Step 1. Since the supporting cone of P from a point can be constructed in time $O(\log n)$ using binary search [48], the running time of the algorithm is $O((k-1)n + (m-k+1)\log n)$, i.e., $O(mn)$ in the worst case, where k is the value of variable i when leaving Step 2. However, the experiments in Section 2.3 will demonstrate that $k = 3$ almost always holds, hence Step 2 will almost never get executed more than once, reducing the running time to $O(n + m \log n)$.

Since a convex m-gon Q is naturally the intersection of m cones, each with a vertex of Q as its vertex and with the internal angle at that vertex as its apex angle, we can use Algorithm 2 to compute all possible inscriptions of P in Q with the same time cost. This problem shall be called the *polygon inscription* problem, which can be regarded as another version of multi-cone inscription because the intersection of multiple cones is always a polygon (possibly unbounded or empty).

2.3 Experiments

The first set of experiments were conducted to find out how many possible poses usually exist for a polygon embedded in three half-planes with non-concurrent bounding lines. The results are summarized in Table 1.

# tests	data source	# poly vertices		# possible poses	
		range	mean	range	mean
10000	10 sq.	3–9	5.9376	1–16	5.5436
10000	10 cir.	3–9	6.1208	2–16	5.7006
1000	100 sq.	6–19	11.917	2–16	7.242
1000	100 cir.	10–22	15.108	2–24	10.012
1000	1000 sq.	11–24	18.43	2–18	7.84
1000	1000 cir.	23–43	33.595	2–60	18.709
1000	cir. mar.	3–15	8.98	2–14	4.537

Table 1: Experiments on embedding a polygon in three half-planes.

Seven groups of convex polygons were tested. The first six groups consisted of convex hulls generated over 10, 100 and 1000 random points successively, and for each number, in two kinds of uniform distributions—inside a square and inside a circle respectively. It can be seen in the table that the polygons in these groups had a wide range (3–43) of sizes, i.e., numbers of vertices, but their shapes were not arbitrary enough, approaching either a square or a circle when large number of random points were used. So we introduced the last group of data that consisted of polygons generated by a method called *circular march* which outputs the vertices of a convex polygon as random points inside a circle in counterclockwise order. The size of a polygon in this group was randomly chosen between 3 and 15.

Given a convex polygon, three supporting lines, each bounding a half-plane on the side of the polygon, were generated according to the uniform distribution; namely, with probability $\frac{\pi-\varphi_i}{2\pi}$ each line passed through vertex p_i with internal angle φ_i. An additional check was performed to ensure that these lines were not attached to the same vertex of the polygon. The number of possible poses for the polygon to be embedded in these generated half-planes was then computed, and the summarized results for all group are listed in the last two columns of the table.

Table 1 tells us that three half-planes are *insufficient* to limit all possible poses of an embedded polygon to a unique one, namely, the real pose; in fact the table suggests that

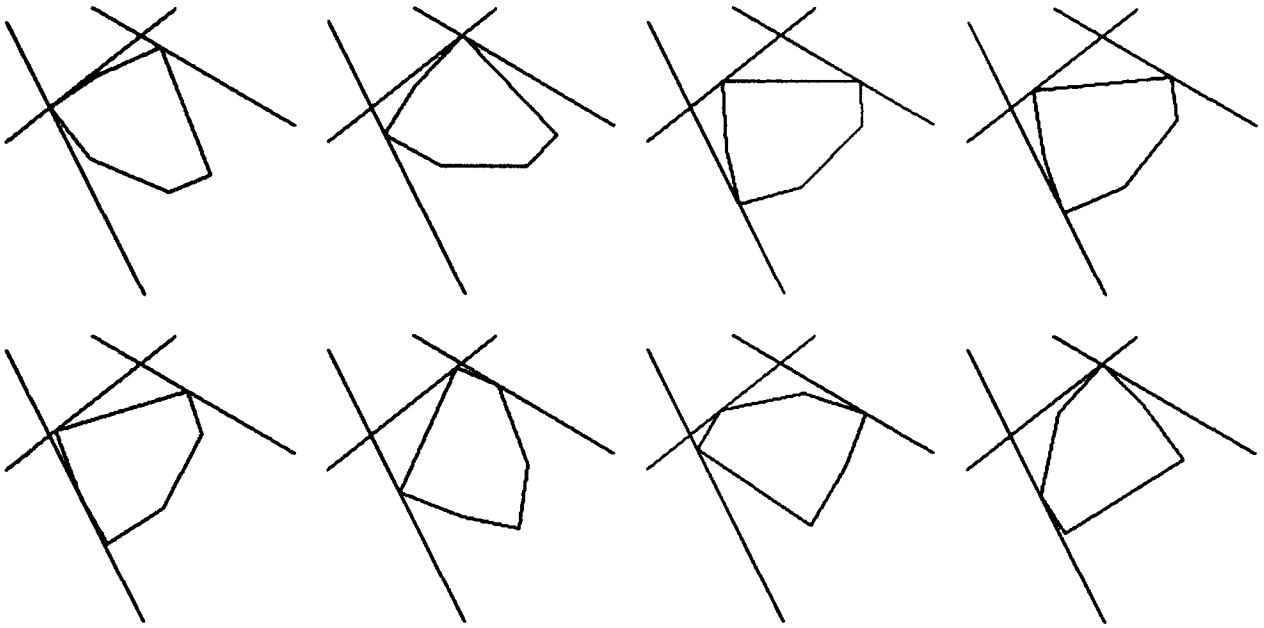


Figure 5: Eight possible poses for a convex 6-gon bounded by three supporting lines (as taken from a sample run). The first one represents the real pose whose supporting lines as shown were generated randomly; the remaining seven represent all other poses consistent with the supporting line constraints. Notice in this example that three of the eight poses occurred when a vertex of the polygon coincided with an intersection of two supporting lines.

linear (in the size of the polygon) number of possible poses will usually exist. We can see in the table that despite the appearances of cases with one or two possible poses, the ratio between the mean of numbers of possible poses and mean polygon size lies in the approximate range 0.43–0.93, decreasing very slowly as the mean polygon size in a group increases. These results tend to support a conjecture that in the average case there exist $O(n)$ possible poses for a convex n-gon embedded in three half-planes with non-concurrent bounding lines.

The above conjecture may be very difficult to prove. However, a plausible explanation for the experimental results can be sought. Recall, a polygon with three half-planes defines a partition of orientation space $[0, 2\pi)$ into at most $3n$ intervals, each containing orientations that would allow the same three incident vertices whenever a possible pose exists at that orientation. The feasible orientations in each interval occur when one supporting line crosses the locus curve of its associated incident vertex. The locus curve results from moving the other two incident vertices along their supporting lines. As these curves (for all intervals) may often cluster together, the likelihood that they get crossed $O(n)$ times in total by the first supporting line is quite large. This happened particularly often during the experiments when a vertex coincided with an intersections of two supporting lines. (See Figure 5.)

The purpose of the second set of experiments was to study how many poses usually exist for a convex polygon inscribed into two or more cones with distinct vertices. We first tested with two cones using the same source of random data generated in the way we did for the first set of experiments, and the results are shown in Table 2. Since a polygon was always generated inside a square (circle), the cone vertices were chosen as random points uniformly

# tests	data source	# poly vertices		# possible poses	
		range	mean	range	mean
10000	10 sq.	3–10	5.9493	1–2	1.036
10000	10 cir.	3–10	6.1113	1–2	1.0117
1000	100 sq.	6–18	12.002	1–2	1.01
1000	100 cir.	9–21	15.138	1–1	1
1000	1000 sq.	10–26	18.073	1–2	1.002
1000	1000 cir.	26–45	33.665	1–2	1.003
1000	cir. mar.	3–15	9.009	1–2	1.106

Table 2: Experiments on inscribing polygons with two cones.

distributed between this boundary and a larger square (circle). The ratio between these two squares (circles) was set uniformly to be $\frac{1}{2}$ for all seven groups of data.

In contrast to Table 1, Table 2 tells us that two cones allow a *unique* pose of an inscribed polygon in most cases. In each group of tests, *only* cases with one pose or two poses occurred, and the mean of possible poses stayed very close to 1, independent of the mean polygon size. (It is not hard for us to see that the percentage of two-pose cases was very low in the range 0%–3.6% for the first six groups of data. The pecentage 10.6% for the seventh group was a bit high but expected, because polygons generated by circular march were more likely to be in a certain shape that would often incur two possible poses, as we will discuss later.)

Tests were also conducted with 3–10 cones on reproduced data of four of the seven groups, while the other experiment parameters were kept the same. As shown in Table 3, the means of possible poses did not decrease dramatically as compared to those in Table 2. Finally,

# tests	data source	# poly vertices		# possible poses	
		range	mean	range	mean
10000	10 sq.	3–10	5.95	1–2	1.0056
1000	100 sq.	6–19	11.901	1–1	1
1000	1000 sq.	10–27	18.26	1–1	1
1000	cir. mar.	3–15	8.96	1–2	1.022
10000	10 sq.	3–10	5.9587	1–2	1.1158
10000	10 sq.	3–10	5.9741	1–2	1.1738

Table 3: More experiments on inscribing polygons with cones. The first four groups of data were tested with a random number (between 3 and 10) of cones; the last two groups were tested with two cones with vertices chosen from wider ranges—the ratio between the two squares (circles) determining the range of cone vertices was set to $\frac{1}{5}$ and $\frac{1}{10}$ respectively for these two groups.

we repeated the first group of tests with two cones but chose their vertices from two wider ranges (with the previous ratio $\frac{1}{2}$ replaced by ratios $\frac{1}{5}$ and $\frac{1}{10}$ respectively), and the results

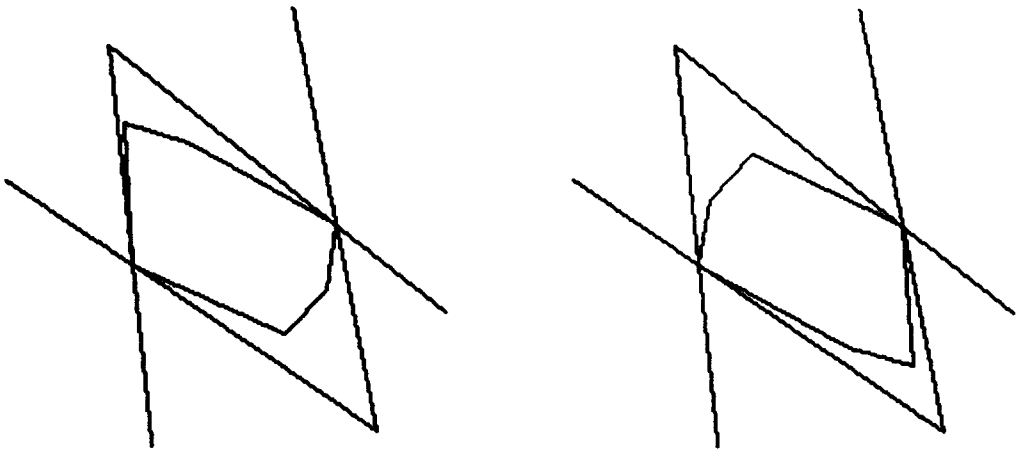


Figure 6: Two possible poses for a convex 6-gon inscribed in two cones (as taken from a sample run). The two cones are supporting the polygon at a pair of vertices.

are shown in the last two rows of the same table.

The experimental result that two non-incident cones usually allow a unique pose of an inscribed convex n-gon P has in fact a very intuitive explanation. As mentioned before, two such cones generally provide four half-planes, any three of which will limit the number of possible poses of P to at most $6n$. Let polygons $P_1, \ldots, P_m$, where $m \leq 6n$, represent P in all possible poses respectively when embedded in the first three half-planes; then those P_i corresponding to the final possible poses must be supported by the bounding line l of the fourth half-plane. So the probability that a two-pose case occurs is no more than the probability that l passes through a vertex of P_i and a vertex of P_j, for $i \neq j$. Note the vertices of $P_1, \ldots, P_m$ together occupy $\Theta(mn)$ points in the plane in general, only one of which must lie on l. If no two of these vertices coincide, the probability that l passes through two vertices of different polygons is zero (assuming that l is independent of the other three bounding lines), which means that a two-pose case almost never occurs in this situation. Otherwise suppose two vertices of P_i and P_j respectively are at the same point p for some $i \neq j$, then the probability that l passes through p is $\Theta(\frac{1}{nm})$. This is $\Theta(\frac{1}{n^2})$ in the average case, as suggested that $m = \Theta(n)$ by the first set of experiments. Since in the usual case there only exist a constant number of such coincident vertex pairs, the probability $\Theta(\frac{1}{n^2})$ is an approximate upper bound on how often two-pose cases occur. This bound turns out to be consistent with the percentages of two-pose cases in Table 2.

It was observed during the experiments that a large number of two-pose cases occurred when both cones happened to be supporting the polygon at the same pair of vertices. (See Figure 6.) The two possible orientations differed by π, and each supporting vertex in one pose coincided with the other in the other pose. This situation often happened when the distance between one pair of vertices of the polygon was much larger than the distance between any other pair of vertices, or when the sites were far away from the polygon (as evidenced by the high percentages of two-pose cases in the last two groups of tests in Table 3).

3 Sensing by Point Sampling

3.1 The Point Sampling Problem

Often the possible poses in which a part settles on the assembly table are not continuous but of a small number, either reduced by a parts feeder or limited by the mechanical constraints imposed by a sequence of planned manipulations. (See Erdmann and Mason's tray-tilting method [22] and Brost's squeeze-grasping method [10] for examples of the latter case.) These facts together allow the implementation of more effective sensing mechanisms. The efficiency of such sensing mechanisms depends on both the time cost of the physical operations and the time complexity of the algorithms involved. Consequently, minimizing one or both of these two factors has become an important aspect of sensor design.

In order to illustrate the goals of this section, consider a polygonal part resting on a horizontal assembly table. The table is bounded by vertical fences at its bottom-left corner, as shown in Figure 7. Pushing the part towards that corner will eventually cause the part to settle in one of the 12 stable poses listed in the figure. (Note to reach a stable pose both fences must be in contact with some vertices of the part while at least one fence must be in contact with no less than two vertices.) In order to distinguish between these 12 poses, the robot has marked 4 points on the table beforehand, so it can infer the pose from which marks are covered by the part and which are not[2].

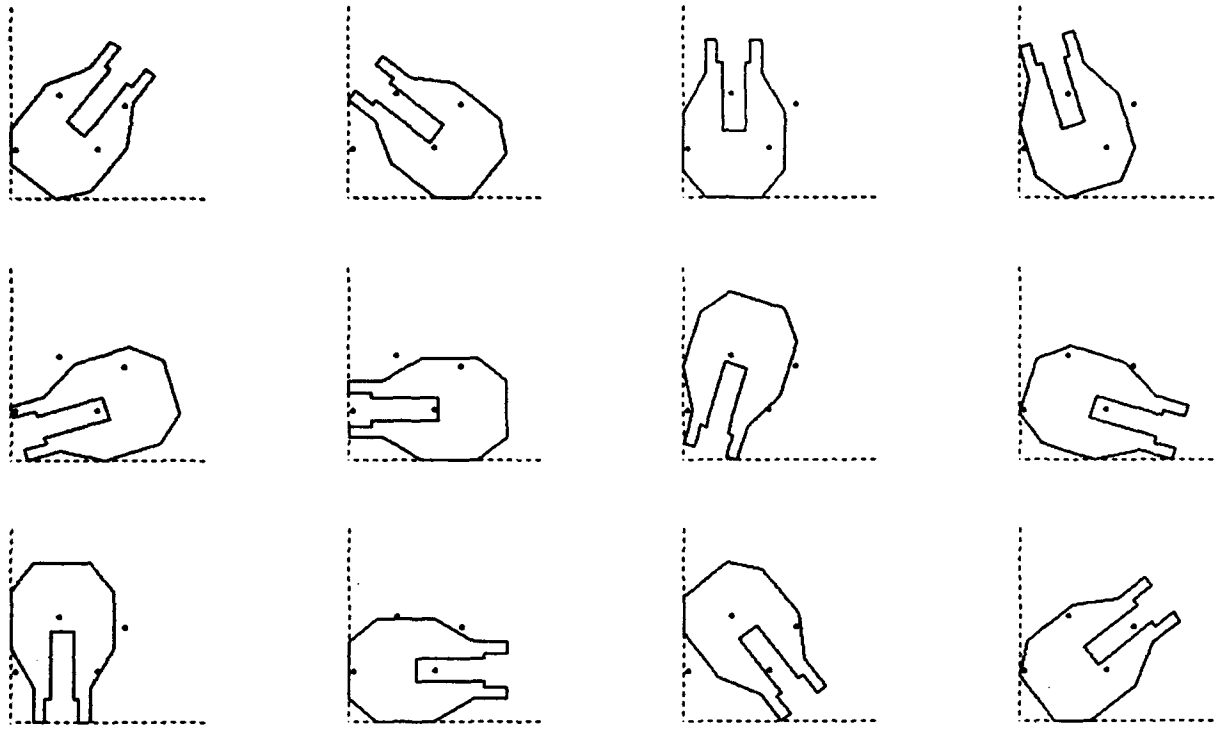
The above "shape recovery" method is named *sensing by point sampling*, as a loose analogy to the reconstruction of band limited functions by sampling on a regular grid in signal processing. To save the expense of sampling, the robot wants to mark as *few* points as possible. The problem: *How to compute a minimum set of points to be marked so that parts of different types and poses can be distinguished from each other by this method?*

3.1.1 Related Work

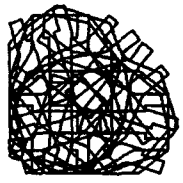
Orienting mechanical parts was early studied by Grossman and Blasgen [31]. They used a vibrating box to constrain a part to a small finite number of possible stable poses, and then determined the particular pose by a sequence of probes using a tactile sensor. Natarajan [46] examined a similar strategy of detecting the orientations of polygonal and polyhedral objects with an analysis of the numbers of sensors sufficient and necessary for the task.

More recent related work includes [8] and [3]. [8] shows that the problem of deciding whether k line probes are sufficient to distinguish a convex polygon from a collection of n convex polygons is NP-complete. This result is very similar to our Theorem 3. A variation of the line-probing result in [8] would give us the point sampling result of Theorem 3. [3] proves a similar result as well, namely that the problem of constructing a decision tree of minimum height to distinguish among n polygons using point probes is NP-complete. This result holds even if all the polygons are convex. [3] also exhibits a greedy approximation algorithm for constructing such a decision tree. This result is similar to our approximation algorithm of Section 3.4, with a similar ratio bound. The difference is that our greedy algorithm seeks to

[2]This can be implemented in multiple ways, such as placing light detectors in the table, probing at the points, or if the robot has a vision system, taking a scene image and checking the corresponding pixel values.



(a)



(b)

Figure 7: Sensing by point sampling. (a) The 12 possible stable poses of an assembly part after pushing, along with 4 sampling points (optimal by Lemma 1) found by our implementation of the approximation algorithm to recognize these poses, where dashed line segments are fences perpendicular to the plane. (b) The planar subdivision formed by these poses which consists of 610 regions.

minimize the total number of probe points rather than the tree height.[3]

Closely related work includes the research by Romanik and others on geometric testability (see, for example, [50], [51], and [1]). Their research develops strategies for verifying a given polygon using a series of point probes. Moreover, the research examines the testability of more general geometric objects, such as polyhedra, and develops conditions that determine whether a class of objects is (approximately) testable.

A number of researchers have looked at the problem of determining or distinguishing objects using finger probes. Finger probing is closely related to sensing by point sampling, as indicated by our discussion of [8]. For a more extensive survey of probing problems and solutions see the paper by Skiena [54].

There would seem to be connections between our work and the concept of VC-dimension often used in learning theory. For instance, in this section we develop the notion of a "discriminating set" to distinguish different polygons. The concept of a discriminating set bears some resemblance to the idea of shattered sets associated with VC-dimension. However, discriminating sets and shattered sets are different. A minimum discriminating set is the smallest set of points that uniquely identifies every object in a set of objects, whereas VC-dimension is the size of the largest set of points shattered by the set of objects. Thus, the VC-dimension of a finite class gives a lower bound on the size of a minimum discriminating set. For dense polygon distributions, the two cardinalities will be the same, namely $\log n$, where n is the number of polygons. For sparsely distributed polygons, the two cardinalities are different. For instance, the VC-dimension can be 1, while the minimum discriminating set has size $n-1$. See Figure 8.

Finally, the work described in this section is part of our larger research goal to understand the information requirements of robot tasks. Related work includes the sensor design methodology of Erdmann [21] and the information invariants of Donald et al. [19]. [21] proposes a method for designing sensors based on the particular manipulation task at hand. The resulting sensors satisfy a minimality property with respect to the given task goal and the available robot actions. [19] investigates the relationship between sensing, action, distributed resources, communication paths, and computation, in the solution of robot tasks. That work provides a method for comparing disparate sensing strategies, and thus for developing minimal or redundant strategies, as desired.

3.1.2 The Formal Problem

Consider n simple polygons $P_1, \ldots, P_n$ in the plane, not necessarily disjoint from each other. We wish to locate the minimum number of points in the plane such that no two polygons P_i and P_j, $i \neq j$, contain exactly the same points. In order to avoid ambiguities in sensing, we require that none of the located points lie on any edge of $P_1, \ldots, P_n$. The planar subdivision formed by $P_1, \ldots, P_n$ divides the plane into one unbounded region, some bounded regions

[3]It is easy to give an example for which a minimum height decision tree uses more than minimum number of total probes, while a decision tree with minimum number of total probes does not attain the minimum height. Consider the problem of discriminating sets $\{a, b, a', e'\}$, $\{a, a'\}$, $\{b, b', d'\}$, $\{c, b'\}$, $\{d, c'\}$ and $\emptyset$ which can be viewed as probing a collection of polygons by the later transformation technique in the proof of Theorem 3. The decision tree using minimum probes a, b, c, d always has height 4, and the decision tree using probes a', b', c', d', e' can achieve minimum height 3.

outside $P_1, \ldots, P_n$, called the "holes", and some bounded regions inside. (For example, the 12 polygons in Figure 7(a) form the subdivision in Figure 7(b) which consists of 610 regions, none of which is a hole.) Immediately we make two observations: (1) Points on the edges of the subdivision or in the interior of the unbounded region or in a "hole" do not need to be considered as locations; (2) for each bounded (open) region inside some polygon only one point needs to be considered.

Let Ω denote the set of bounded regions in the subdivision which are contained in at least one of $P_1, \ldots, P_n$. Each polygon P_i, $1 \leq i \leq n$, is partitioned into one or more such regions; we write $\omega \sqsubseteq P_i$ when a region ω is contained in polygon P_i. A *region basis* for polygons $P_1, \ldots, P_n$ is a subset $\Delta \subseteq \Omega$ such that

$$\{\omega \mid \omega \in \Delta \text{ and } \omega \sqsubseteq P_i\} \neq \{\omega \mid \omega \in \Delta \text{ and } \omega \sqsubseteq P_j\},$$

for $1 \leq i \neq j \leq n$; that is, each P_i contains a distinct collection of regions from Δ. A region basis Δ^* of minimum cardinality is called a *minimum region basis.* Thus the problem of sensing by point sampling becomes the problem of finding a minimum region basis Δ^*. We will call this problem *Region Basis* and focus on it throughout the section. The following lemma gives the upper and lower bounds for the size of such Δ^*.

Lemma 1 *A minimum region basis Δ^* for n polygons $P_1, \ldots, P_n$ satisfies $\lceil \log n \rceil \leq |\Delta^*| \leq n - 1$.*

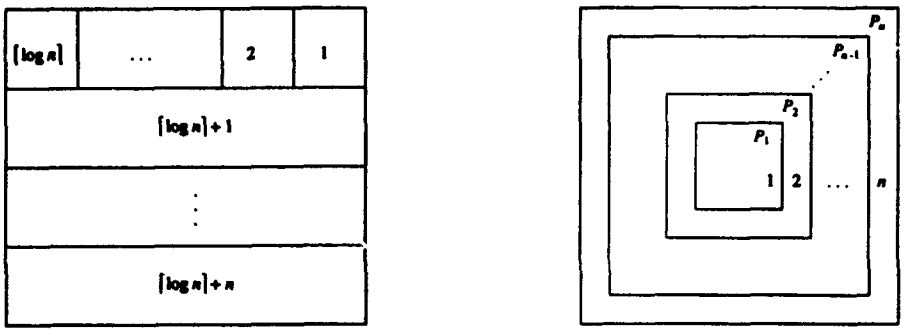
Proof. To verify the lower bound $\lceil \log n \rceil$, note that each of the n polygons must contain a distinct subset of Δ^*; so $n \leq 2^{|\Delta^*|}$, the cardinality of the power set 2^{Δ^*}.

To verify the upper bound $n - 1$, we incrementally construct a region basis Δ of size at most $n - 1$. This construction is similar to Natarajan's Algorithm 2 [46]. Initially, $\Delta = \emptyset$. If $n > 1$, without loss of generality, assume P_1 has the *smallest area.* Then there exists some region $\omega_1 \in \Omega$ outside P_1. Split $\{P_1, \ldots, P_n\}$ into two *nonempty* subsets, one including those P_i containing ω_1 and the other including those not; and add ω_1 into Δ. Recursively split the resulting subsets in the same way, and at each split, add into Δ its defining region (as we did with ω_1) if this region is not already in Δ, until every subset eventually becomes a singleton. The Δ thus formed is a region basis. Since there are $n - 1$ splits in total and each split adds at most one region into Δ, we have $|\Delta| \leq n - 1$. □

Figure 8 gives two examples for which $|\Delta^*| = \lceil \log n \rceil$ and $|\Delta^*| = n - 1$ respectively. Therefore these two bounds are tight.

We can view all the bounded non-hole regions as elements of Ω, and all the polygons $P_1, \ldots, P_n$ as subsets of Ω. Then a region basis Δ is a subset of Ω that can discriminate subsets $P_1, \ldots, P_n$ by intersection. Hence the Region Basis problem can be rephrased as: Find a subset of Ω of minimum size whose intersections with any two subsets P_i and P_j, $1 \leq i \neq j \leq n$, are not equal. The general version of this set-theoretic problem, in which Ω stands for an arbitrary finite set and $P_1, \ldots, P_n$ stand for arbitrary subsets of Ω, we call *Discriminating Set.* We have thus reduced Region Basis to Discriminating Set, and the former problem will be solved once we solve the latter one.

Let us analyze the amount of computation required for the geometric preprocessing to reduce Region Basis to Discriminating Set. Let m be the total *size* of $P_1, \ldots, P_n$, i.e., the sum



(a) (b)

Figure 8: Two examples whose minimum region basis sizes achieve the lower bound $\lceil \log n \rceil$ and the upper bound $n-1$ respectively. Bounded regions in the examples are labelled with numbers. (a) For $1 \leq i \leq n$ polygon P_i is defined to be the boundary of the union of regions $\lceil \log n \rceil + 1, \ldots, \lceil \log n \rceil + i$, and all regions k with $1 \leq k \leq \lceil \log n \rceil$ such that the kth bit of the binary representation (radix 2) for $i-1$ is 1. Thus $\Delta^* = \{1, 2, \ldots, \lceil \log n \rceil\}$. (b) The polygons $P_1, \ldots, P_n$ contain each other in increasing order: $\Delta^* = \{2, 3, \ldots, n\}$.

of the number of vertices each polygon has; trivially $m \geq 3$. Then the planar subdivision these polygons define has at most s vertices, where $3 \leq s \leq \binom{m}{2}$. By Euler's relation on planar graphs, the number of regions and the number of edges are upper bounded by $2s-4$ and $3s-6$ respectively. So we can construct the planar subdivision either in time $O(m \log m + s)$ using an optimal algorithm for intersecting line segments by Chazelle and Edelsbrunner [13], or in time $O(s \log m)$ using a simpler plane sweep version by Nievergelt and Preparata [47]. To obtain the set of regions each polygon contains, we only need to traverse the portion of the subdivision bounded by that polygon, which takes time $O(s)$. It follows that the reduction to Discriminating Set can be done in time $O(m \log m + ns)$, or $O(nm^2)$ in the worst case.

Here is a short summary of the structure of Section 3: Section 3.2 proves the NP-completeness of Discriminating Set; based on this result, Section 3.3 establishes an equivalence between Discriminating Set and Region Basis, hence proving the latter problem NP-complete; Section 3.4 presents an $O(n^2m^2)$ approximation algorithm for Region Basis with ratio $2 \ln n$ and shows that further improvements on this ratio are hard; and Section 3.5 closes up with a complexity analysis of various subproblems of Discriminating Set, along with the definition of a family of related NP-complete problems called k-Independent Sets. We have implemented our approximation algorithm and have tested it on both real data taken from mechanical parts and random data extracted from the arrangements of random lines. The algorithm works very well in practice.

3.2 Discriminating Set

Given a collection C of subsets of a finite set X, suppose we want to identify these subsets just from their *intersections* with some subset $D \subseteq X$. Thus D must have distinct intersection

with every member of C, that is,

$$D \cap S \neq D \cap T, \qquad \text{for all } S, T \in C \text{ and } S \neq T.$$

We call such a subset D a *discriminating set* for C with respect to X. From a different point of view, each element $x \in D$ can be regarded as a binary "bit" that, to represent any subset $S \subseteq X$, gives value '1' if $x \in S$ and value '0' otherwise. In such a way D represents an encoding scheme for subsets in C.

Below we show that the problem of finding a minimum discriminating set is NP-complete. As usual, we consider the decision version of this minimization problem:

DISCRIMINATING SET (D-SET)
Let C be a collection of subsets of a finite set X and $l \leq |X|$ a non-negative integer. Is there a discriminating set $D \subseteq X$ for C such that $|D| \leq l$?

Our proof of the NP-completeness for D-Set uses a reduction from Vertex Cover (VC) which determines if a graph $G = (V, E)$ has a *cover* of size not exceeding some integer $l \geq 0$, i.e., a subset $V' \subseteq V$ that, for each edge $(u, v) \in E$, contains either u or v. The reduction is based on a key observation, that for any three finite sets S_1, S_2 and S_3,

$$S_1 \cap S_2 \neq S_1 \cap S_3 \quad \text{if and only if} \quad S_1 \cap (S_2 \triangle S_3) \neq \emptyset,$$

where '$\triangle$' denotes the operation of symmetric difference, i.e., $S_2 \triangle S_3 = (S_2 \setminus S_3) \cup (S_3 \setminus S_2)$.

Theorem 2 *Discriminating Set is NP-complete.*

Proof. That D-Set $\in$ NP is trivial.

Next we establish VC $\propto_P$ D-Set, that is, there exists a polynomial-time reduction from VC to D-Set. Let $G = (V, E)$ and integer $0 < l \leq |V|$ be an instance of VC. We need to construct a D-Set instance (X, C) such that the collection C of subsets of X has a discriminating set of size l' or less if and only if G has a vertex cover of size l or less.

The construction uses the component design technique described by Garey and Johnson [25]. It's rather natural for us to begin by including every vertex of G in set X, and assigning each edge $e = (u, v)$ a subset $S(e)$ in C which contains at least u and v; in other words, we have $V \subset X$ and

$$\{u, v\} \subset S(e) \in C, \qquad \text{for all } e = (u, v) \in E.$$

In order to ensure that any discriminating set D for C contains at least one of u and v from subset $S(e)$, we add an *auxiliary* subset A_e into C that consists of some new elements *not* in V, and in the meantime define

$$S(e) = \{u, v\} \cup A_e.$$

Hence $S(e) \triangle A_e = \{u, v\}$; and $D \cap \{u, v\} \neq \emptyset$ follows directly from $D \cap S(e) \neq D \cap A_e$. Since any discriminating set D' for $\{ A_e \mid e \in E \}$ can also distinguish between $S(e_1)$ and $S(e_2)$, and between $S(e_1)$ and A_{e_2}, for any $e_1, e_2 \in E$ and $e_1 \neq e_2$, D' unioned with a vertex cover for G becomes a discriminating set for C. Conversely, every discriminating set D for C can be split into a discriminating set for $\{ A_e \mid e \in E \}$ and a vertex cover for G.

The $m = |E|$ auxiliary subsets should be constructed in a way such that we can easily determine the size of their minimum discriminating sets in order to set up the entire D-Set instance. There is a simple way: We introduce m elements $a_1, a_2, \ldots, a_m \notin V$ into X, and define subsets A_e, for $e \in E$, to be

$$\{a_1\},\ \{a_2\},\ \ldots,\ \{a_m\},$$

where the order of mapping does not matter. It's clear that there are m minimum discriminating sets for the above subsets: $\{a_1, \ldots, a_m\} \setminus \{a_i\}$, $1 \leq i \leq m$.

Setting $l' = l + m - 1$, we have completed our construction of the D-Set instance as

$$\begin{aligned} X &= V \cup \{a_1, \ldots, a_m\}; \qquad a_1, \ldots, a_m \notin V; \\ C &= \{\, S(e) \mid e \in E \,\} \cup \{\, A_e \mid e \in E \,\}. \end{aligned}$$

The construction can be carried out in time $O(|V| + |E|)$. We omit the remaining task of verifying that G has a vertex cover of size at most l if and only if C has a discriminating set of size at most $l + m - 1$. □

One thing about this proof is worthy of note. All subsets in C constructed above have at most three elements. This reveals that D-Set is still NP-complete even if $|S| \leq 3$ for all $S \in C$, a stronger assertion than Theorem 2. The subproblem where all $S \in C$ have $|S| \leq 1$ is obviously in P, for an algorithm can simply count $|C|$ in linear time and then answer "yes" if $l \geq |C| - 1$ and "no" if $0 \leq l < |C| - 1$. For the remaining case in which all $S \in C$ have $|S| \leq 2$, we will prove in Section 3.5 that the NP-completeness still holds. However, the proof will be a bit more involved than the one we just gave under no restriction on $|S|$.

At the end of Section 3.2, we give a problem that is equivalent to D-Set:

ROW-DIFFERING SUBMATRIX
Given an $m \times n$ matrix A of 0's and 1's and integer $0 \leq l \leq n$, is there an $m \times l$ matrix B formed by l columns of matrix A such that no two rows of B are identical?

3.3 Region Basis

Now that we have shown the NP-completeness of D-Set, the minimum region basis cannot be computed in polynomial time through the use of an efficient algorithm for D-Set, because no such algorithm would exist unless P = NP. This conclusion, nevertheless, leads us to conjecture that the minimization problem Region Basis is also NP-complete. Again we consider the decision version:

REGION BASIS (RB)
Given n polygons $P_1, \ldots, P_n$ and integer $0 \leq l \leq n - 1$, does there exist a region basis Δ for the planar subdivision Ω formed by $P_1, \ldots, P_n$ such that $|\Delta| \leq l$?

The condition $0 \leq l \leq n - 1$ above is necessary because we already know from Lemma 1 that a minimum region basis has size at most $n - 1$.

Consider a mapping $\mathcal{F}$ from the set of RB instances to the set of D-Set instances that maps regions to elements and polygons to subsets in a one-to-one manner. Every RB instance

is thus mapped into an equivalent D-Set instance, as pointed out in Section 3.1. We claim that $\mathcal{F}$ is not onto. Suppose $\mathcal{F}$ were onto. Then the elements of each subset in a D-Set instance must correspond to regions in some RB instance. The union of these regions must be a polygon, and this polygon must map to the subset given in the D-Set instance. However, this is not always possible. Consider a D-Set instance generated from a nonplanar graph such that each edge is a subset containing its two vertices as only elements. No RB instance can be mapped to such a D-Set instance. For if there were such an RB instance, the geometric dual of the planar subdivision it defines would contain a planar embedding for the original nonplanar graph. This is an impossibility, hence we have a contradiction.

Thus the set of RB instances constitutes a *proper* subset of the set of D-Set instances; in other words, RB is isomorphic to a subproblem of D-Set. Therefore, the NP-completeness of RB does not follow directly from that of D-Set established earlier. Fortunately, however, D-Set has an equivalent subproblem which is isomorphic to a subproblem of RB under $\mathcal{F}$. That isomorphism provides us with the NP-completeness of Region Basis.

Theorem 3 *Region Basis is NP-complete.*

Proof. That RB $\in$ NP is easy to verify, based on the fact mentioned in Section 3.1 that the number of regions in the planar subdivision is at most quadratic in the total size of the polygons.

Let (X, C) be a D-Set instance, where

$$\begin{aligned} X &= \{x_1, x_2, \ldots, x_m\}; \\ C &= \{S_1, S_2, \ldots, S_n\} \subseteq 2^X. \end{aligned}$$

Without loss of generality, we make two assumptions

$$\bigcup_{i=1}^{n} S_i = X \qquad \text{and} \qquad \bigcap_{i=1}^{n} S_i = \emptyset,$$

because elements contained in none of the subsets or contained in all subsets can always be removed from any discriminating set of (X, C). Now add in a new element $a \notin X$ and consider the D-Set instance $(X \cup \{a\}, C')$, where $C' = \{ S_i \cup \{a\} \mid 1 \leq i \leq n \}$. Clearly $(X \cup \{a\}, C')$ and (X, C) have the same set of irreducible discriminating sets[4] and hence they are considered equivalent.

The planar subdivision defined by the constructed RB instance for $(X \cup \{a\}, C')$ takes the configuration shown in Figure 9(a): A rectangular region is divided by a horizontal line segment into two identical regions of which the bottom region is named $\omega(a)$; the top region is further divided, this time by vertical line segments, into $2m - 1$ identical regions of which the odd numbered ones, from left to right, are named $\omega(x_1), \ldots, \omega(x_m)$ respectively. Remove those $m-1$ unnamed regions on the top. For $1 \leq i \leq n$ define polygon P_i to be the boundary of the union of all regions $\omega(x)$, $x \in S_i \cup \{a\}$. It should be clear that P_i is indeed a polygon; and the two assumptions guarantee that $P_1, \ldots, P_n$ form the desired subdivision. Note the subdivision consists of $m + 1$ rectangular regions and $4m + 2$ vertices. All can be computed

[4] A discriminating set D is said to be *irreducible* if no subset $D' \subset D$ can be a discriminating set.

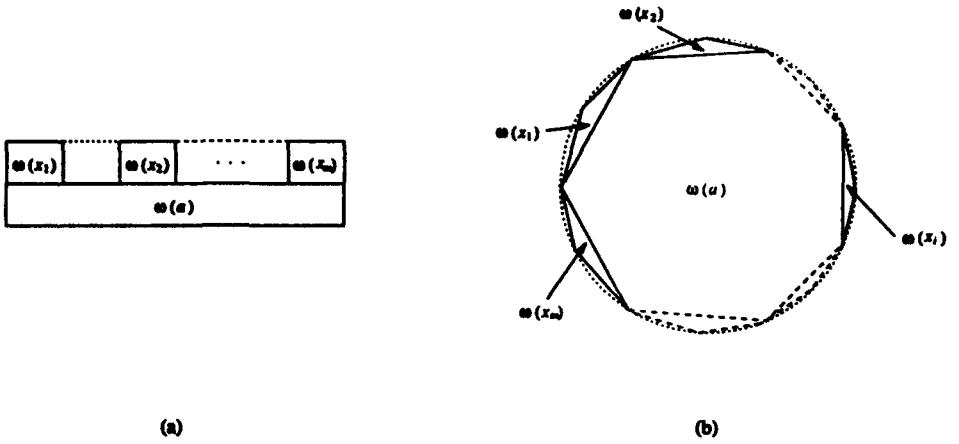


Figure 9: Two reductions from Discriminating Set to Region Basis.

in time $\Theta(m)$, given the coordinates of the four vertices of the bounding rectangle. Thus the reduction takes time $\Theta(\sum_{i=1}^{n} |S_i|)$.

It is clear that C has a discriminating set of size l or less if and only if there is a region basis of the same size for $P_1, \ldots, P_n$. Hence we have proved the NP-completeness of RB. □

The above proof implies that we may regard Discriminating Set and Region Basis as equivalent problems. Note that the polygons $P_1, \ldots, P_n$ in Figure 9 are not convex; will Region Basis become P when all the polygons are convex? This question is answered by the following corollary.

Corollary 2 *Region Basis remains NP-complete even if all the polygons are convex.*

Proof. Same as the proof of Theorem 3 except that we use the planar subdivision shown in Figure 9(b). (The vertices of the subdivision partition an imaginary circle (dotted) into $2n$ equal arcs.) □

3.4 Approximation

Sometimes we can derive a polynomial-time approximation algorithm for the NP-complete problem at hand from some existing approximation algorithm for another NP-complete problem, by reducing one problem to the other. In fewer cases, where the reduction *preserves* the solutions, namely, every instance of the original problem and its reduced instance have the same set of solutions, any approximation algorithm for the reduced problem together with the reduction will solve the original problem. The problem to which we will reduce Discriminating Set is Hitting Set:

HITTING SET
Given a collection C of subsets of a finite set X, find a minimum *hitting set* for C, i.e., a subset $H \subseteq X$ of minimum cardinality such that $H \cap S \neq \emptyset$ for all $S \in C$?

Karp [36] shows Hitting Set to be NP-complete by a reduction from Vertex Cover. The reducibility from D-Set to Hitting Set follows a key fact we observed when proving Theorem 2:

The intersections of a finite set D with two finite sets S and T are not equal if and only if D intersects their symmetric difference $S \Delta T$. Given a D-Set instance, the corresponding Hitting Set instance is constructed simply by replacing all the subsets with their *pairwise* symmetric differences. Thus every discriminating set of the original D-Set instance is also a hitting set of the constructed instance, and vice versa.

The approximability of Hitting Set can be studied through another problem, Set Covering:

SET COVERING
Given a collection C of subsets of a finite set X, find a minimum *cover* for X, i.e., a subcollection $C' \subseteq C$ of minimum size such that $\bigcup_{S \in C'} S = X$?

This problem is also shown to be NP-complete by Karp by a reduction from Exact Cover by 3-Sets [36]. A greedy approximation algorithm for this problem due to Johnson [34] and Lovász [43] guarantees to find a cover $\hat{C}$ for X with ratio

$$\frac{|\hat{C}|}{|C^*|} \leq H(\max_{S \in C} |S|) \quad \text{or simply} \quad \frac{|\hat{C}|}{|C^*|} \leq \ln |X| + 1,$$

where C^* is a minimum cover and $H(k) = H_k = \sum_{i=1}^{k} \frac{1}{i}$, known as the kth harmonic number. The algorithm works by selecting, at each stage, a subset from C that covers the most remaining uncovered elements of X. We refer the reader to [15] for a general analysis of the greedy heuristic for Set Covering.

Hitting Set and Set Covering are *duals* to each other—the roles of set and element in one problem just get switched in the other. More specifically, let a Hitting Set instance consist of some finite set X and a collection C of its subsets; its dual Set Covering instance then consists of a set $\bar{C}$ and a collection of its subsets $\bar{X}$ where

$$\bar{C} = \{ \bar{S} \mid S \in C \} \quad \text{and} \quad \bar{X} = \{ \bar{x} \mid x \in X \},$$

and where each subset $\bar{x}$ is defined as[5,6]

$$\bar{x} = \{ \bar{S} \mid S \in C \text{ and } S \ni x \}.$$

Intuitively speaking, the element $x \in X$ "hits" the subset $S \in C$ in the original instance if and only if the subset $\bar{x}$ "covers" the element $\bar{S} \in \bar{C}$ in the dual instance. Thus it follows that $H \subseteq X$ is a hitting set for C if and only if $\bar{H} = \{ \bar{x} \mid x \in H \}$ is a cover for $\bar{C}$. Hence the corresponding greedy algorithm for Hitting Set selects at each stage an element that "hits" the most remaining subsets. It is clear that the approximation ratio for Hitting Set becomes $H(\max_{x \in X} |\{ S \mid S \in C \text{ and } S \ni x \}|)$ or $\ln |C| + 1$.[7]

[5] According to this definition, $\bar{x} = \bar{y}$ may hold for two different elements $x \neq y$. In this case only one subset is included in $\bar{X}$.

[6] This definition also establishes the duality between D-Set and a known NP-complete problem called Minimum Test Set (see [25]). Given a collection of subsets of a finite set, Minimum Test Set asks for a minimum subcollection such that exactly one from each pair of distinct elements is contained in some subset from this subcollection.

[7] Kolaitis and Thakur [39] syntactically define a class of NP-complete problems with logarithmic approximation algorithms that contains Set Covering and Hitting Set, and show that Set Covering is complete for the class.

As a short summary, the greedy heuristic on a Discriminating Set instance (X, C) works by finding a hitting set for the instance $(X, \{ S \triangle T \mid S, T \in C \})$. Since an element can appear in at most $\lfloor \frac{n^2}{4} \rfloor$ such pairwise symmetric differences, where $n = |C|$, the approximation ratio attained by this heuristic is $\ln \lfloor \frac{n^2}{4} \rfloor + 1 < 2 \ln n$. The same ratio is attained for Region Basis by the heuristic that selects at each step a region discriminating the most remaining pairs of polygons, where n is now the number of polygons.

The greedy algorithm for Set Covering (dually for Hitting Set) can be carefully implemented to run in time $O(\sum_{S \in C} |S|)$ [17]. The reduction from a D-Set instance (X, C) to a Hitting Set instance takes time $O(|C|^2 \max_{S \in C} |S|)$. Combining the time complexity of the geometric preprocessing in Section 3.1, we can easily verify that Region Basis can be solved in time $O(nm^2 + n^2m^2) = O(n^2m^2)$, where n and m are the number and size of polygons respectively.

In the remainder of this subsection we establish the hardness of approximating D-Set and hence Region Basis. Both problems allow the same approximation ratio since the reductions from one to another do not change the number of subsets (or polygons) in an instance. First we should note that the ratio bound $H(\max_{S \in C} |S|)$ of the greedy algorithm for Set Covering is actually tight; an example that makes the algorithm achieve this ratio for arbitrarily large $\max_{S \in C} |S|$ is given in [34].

Next we present a reverse reduction from Hitting Set to D-Set to show that an algorithm for D-Set with approximation ratio $c \log n$ can be used to obtain an algorithm for Hitting Set with ratio $c \log n + O(\log \log n)$, where $c > 0$ is any constant and n is the number of subsets in an instance. Afterwards, we will apply some recent results on the hardness of approximating Set Covering (and thus Hitting Set).

Lemma 2 *For any $c > 0$, if $c \log n$ is the approximation ratio of Discriminating Set, then Hitting Set can be approximated with ratio $c \log n + O(\log \log n)$.*

Proof. Suppose there exists an algorithm $\mathcal{A}$ for D-Set with approximation ratio $c \log n$. Let (X, C) be an arbitrary instance of Hitting Set, where $C = \{S_1, \ldots, S_n\} \subseteq 2^X$, and let $n = |C|$. To construct a D-Set instance, we first make $f(n)$ isomorphic copies $(X_1, C_1), \ldots, (X_{f(n)}, C_{f(n)})$ of (X, C) such that $X_i \cap X_j = \emptyset$ for $1 \leq i \neq j \leq f(n)$. Here f is an as yet undetermined function of n upper bounded by some polynomial in n. Now consider the enlarged Hitting Set instance $(X', C') = (\bigcup_{i=1}^{f(n)} X_i, \bigcup_{i=1}^{f(n)} C_i)$. Every hitting set H' of (X', C') has $H' = \bigcup_{i=1}^{f(n)} H_i$, where H_i is a hitting set of (X_i, C_i), $1 \leq i \leq n$; so from H' we can obtain a hitting set H of (X, C) with $|H| \leq |H'|/f(n)$ merely by taking the smallest one of $H_1, \ldots, H_{f(n)}$.

Next we introduce a set A consisting of new elements $a_1, a_2, \ldots, a_{\log(nf(n))} \notin X'$; and for $1 \leq i \leq nf(n)$ define auxiliary sets A_i:

$$A_i = \{ a_j \mid 1 \leq j \leq \log(nf(n)) \text{ and the } j\text{th bit of the binary representation of } i - 1 \text{ is } 1 \}.$$

It is not hard to see $\{a_1, \ldots, a_{\log(nf(n))}\}$ must be a subset of any discriminating set for $A_1, \ldots, A_{nf(n)}$; therefore it is the minimum one for these auxiliary sets. The constructed D-Set instance is then defined to be (X'', C''), where

$$\begin{aligned} X'' &= X_1 \cup \cdots \cup X_{f(n)} \cup \{a_1, a_2, \ldots, a_{\log(nf(n))}\}; \\ C'' &= \{ T \cup A_{(i-1)n+j} \mid T \in C_i \text{ and } T \cong S_j \} \cup \{ A_1, \ldots, A_{nf(n)} \}. \end{aligned}$$

It is easy to verify that every discriminating set of (X'', C'') is the union of A and a hitting set of (X', C').

Now run algorithm $\mathcal{A}$ on the instance (X'', C'') and let D be the discriminating set found. Then

$$\frac{|D|}{|D^*|} \leq c \log(|C''|) = c \log(2nf(n)),$$

where D^* is a minimum discriminating set. From the construction of (X'', C'') we know that $D = H_1 \cup \cdots \cup H_{f(n)} \cup A$ and $D^* = H_1^* \cup \cdots \cup H_{f(n)}^* \cup A$, where for $1 \leq i \leq n$, H_i and H_i^* are some hitting set and some minimum hitting set of (X_i, C_i), respectively. Let H_k satisfy $|H_k| = \min_{i=1}^{f(n)} |H_i|$ and thus let $H \cong H_k$ be a hitting set of (X, C). Also, let H^* with $|H^*| = |H_1^*| = \cdots = |H_{f(n)}^*|$ be a minimum hitting set of (X, C). Then

$$\begin{aligned} \frac{|D|}{|D^*|} &= \frac{\sum_{i=1}^{f(n)} |H_i| + |A|}{\sum_{i=1}^{f(n)} |H_i^*| + |A|} \\ &\geq \frac{f(n) \cdot |H| + \log(nf(n))}{f(n) \cdot |H^*| + \log(nf(n))}. \end{aligned}$$

Combining the two inequalities above generates:

$$\begin{aligned} \frac{|H|}{|H^*|} &\leq c \log(2nf(n)) + \frac{(c \log(2nf(n)) - 1) \cdot \log(nf(n))}{f(n) \cdot |H^*|} \\ &\leq c \log(2nf(n)) + \frac{(c \log(2nf(n)) - 1) \cdot \log(nf(n))}{f(n)} \\ &= c \log n + \left[c + c \log f(n) + \frac{(c \cdot (1 + \log n + \log f(n)) - 1) \cdot (\log n + \log f(n))}{f(n)} \right]. \end{aligned}$$

Setting $f(n) = \log^2 n$, all terms in the brackets can be absorbed into $O(\log \log n)$ after simple manipulations on asymptotics [29]; thus we have

$$\frac{|H|}{|H^*|} \leq c \log n + O(\log \log n).$$

□

Though Set Covering has been extensively studied since the mid 70's, essentially nothing on the hardness of approximation was known until very recently. The results of [4] imply that no polynomial approximation scheme exists unless P = NP. Based on recent results from interactive proof systems and probabilistically checkable proofs and their connection to approximation, several asymptotic improvements on the hardness of approximating Set Covering have been made. In particular, Lund and Yannakakis [44] showed that Set Covering cannot be approximated with ratio $c \log n$ for any $c < \frac{1}{4}$ unless NP $\subset$ DTIME($n^{\text{poly} \log n}$); Bellare et al. [7] showed that approximating Set Covering within any constant is NP-complete, and approximating it within $c \log n$ for any $c < \frac{1}{8}$ implies NP $\subset$ DTIME($n^{\log \log n}$). Based on their results and by Lemma 2, we conclude on the same hardness of approximating D-Set and Region Basis:

Theorem 4 *Discriminating Set and Region Basis cannot be approximated by a polynomial-time algorithm with ratio bound* $c \log n$ *for any* $c < \frac{1}{4}$ *unless* NP $\subset$ DTIME($n^{\text{poly}\log n}$), *or for any* $c < \frac{1}{8}$ *unless* NP $\subset$ DTIME($n^{\log\log n}$).

Following the above theorem, the ratio $2 \ln n \approx 1.39 \log n$ of the greedy algorithm for D-Set remains asymptotically optimal if NP is not contained in DTIME($n^{\text{poly}\log n}$).

3.5 More on Discriminating Set

Now let's come back to where we left the discussion on the subproblems of D-Set in Section 3.2; it has not been settled whether D-Set remains NP-complete when every subset S in the collection C satisfies $|S| \leq 2$. We now prove that this subproblem is NP-complete.

Here we look at a special case of this subproblem, namely, a "subsubproblem" of D-Set, subject it to two restrictions: (1) $\emptyset \in C$ and (2) $|S| = 2$ for all nonempty subsets $S \in C$. Let's call this special case *0-2 D-Set*. If 0-2 D-Set is proven to be NP-complete, so will be the original subproblem.

It's quite intuitive to understand a 0-2 D-Set instance in terms of a graph $G = (V, E)$, where $V = X$, the finite set of which every $S \in C$ is a subset, and

$$E = \Big\{ (u, v) \;\Big|\; \{u, v\} \in C \Big\}.$$

In other words, each element of the set X corresponds to a vertex in G while each subset, except $\emptyset$, corresponds to an edge. Clearly this correspondence from all 0-2 D-Set instances to all graphs is one-to-one. Since any discriminating set D for C has

$$D \cap S \neq D \cap \emptyset = \emptyset, \qquad \text{for all } S \in C \text{ and } S \neq \emptyset,$$

D must be a vertex cover for G. Let $d(u, v)$ be the *distance*, i.e., the length of the shortest path, between vertices u,v in G (or ∞ if u and v are disconnected). A *3-independent set* in G is a subset $I \subseteq V$ such that $d(u, v) \geq 3$ for every pair $u, v \in I$. The following lemma captures the dual relationship between a discriminating set for C and a 3-independent set in G.

Lemma 3 *Let* X *be a finite set and* C *a collection of* $\emptyset$ *and two-element subsets of* X. *Let* $G = (X, E)$ *be a graph with* $E = \{ (u, v) \mid \{u, v\} \in C \}$. *Then a subset* $D \subseteq X$ *is a discriminating set for* C *if any only if* $X \setminus D$ *is a 3-independent set in* G.

Proof. Let D be a discriminating set for C. Assume there exist two distinct elements (vertices) $u, v \in X \setminus D$ such that $d(u, v) < 3$. We immediately have $(u, v) \notin E$ since D must be a vertex cover in G; so $d(u, v) = 2$. Hence there is a third vertex, say w, that is connected to both u and v; furthermore, $w \in D$ holds since the edges (u, w) and (v, w) must be covered by D. But now we have $D \cap \{u, w\} = D \cap \{v, w\} = \{w\}$, a contradiction to the fact that D is a discriminating set.

Conversely, suppose $X \setminus D$ is a 3-independent set in G, for some subset $D \subseteq X$. Then D must be a vertex cover. Suppose it is not a discriminating set for C. Then there exist two distinct subsets $S_1, S_2 \in C$ such that $D \cap S_1 = D \cap S_2 = \{w\}$, for some $w \in S$. Writing

$S_1 = \{u, w\}$ and $S_2 = \{v, w\}$, we have $d(u, v) = 2$; but in the meantime $u, v \in X \setminus D$. A contradiction again. □

This lemma tells us that the NP-completeness of 0-2 D-Set, and therefore of our remaining open subproblem of D-Set, follows if we can show the NP-completeness of *3-Independent Set*. 3-Independent Set is among a family of problems defined, for all integers $k > 0$, as follows:

k-INDEPENDENT SET (k-IS)
Given a graph $G = (V, E)$ and an integer $0 < l \leq |V|$, is there a ***k-independent set*** of size at least l, that is, is there a subset $I \subseteq V$ with $|I| \geq l$ such that $d(u, v) \geq k$ for every pair $u, v \in I$?

Thus 2-IS is the familiar NP-complete Independent Set problem. We will see in Appendix B that every problem in this family for which $k > 3$ is also NP-complete. To avoid too much divergence from 0-2 D-Set, let's focus on 3-IS only here.

Lemma 4 *3-Independent Set is NP-complete.*

Proof. It is trivial that 3-IS $\in$ NP. To show NP-hardness, we reduce Independent Set (2-IS) to 3-IS. Let $G = (V, E)$ and $0 < l \leq |V|$ form an instance of Independent Set. A graph G' is constructed from G in two steps. In the first step, we introduce a "midvertex" $\alpha_{u,v}$ for each edge $(u, v) \in E$, and replace this edge with two edges $(u, \alpha_{u,v})$ and $(\alpha_{u,v}, v)$. In the second step, an edge is added between every two midvertices that are adjacent to the same original vertex. More formally, we have defined $G' = (V', E')$ where

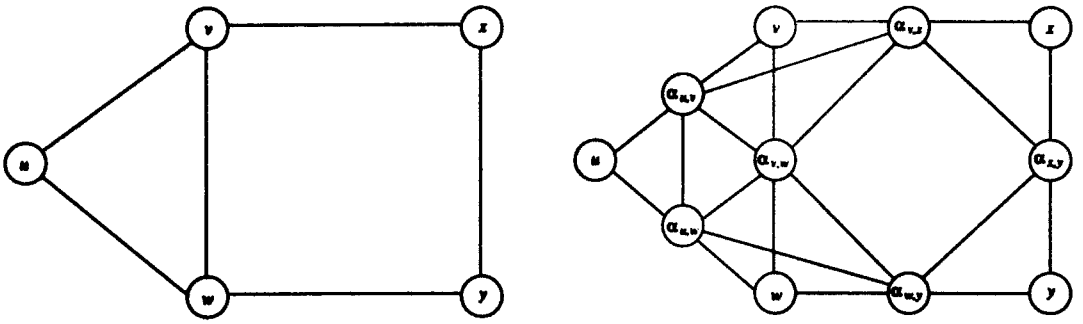
$$\begin{aligned} V' &= V \cup \left\{ \alpha_{u,v} \;\middle|\; (u, v) \in E \right\}; \\ E' &= \left\{ (\alpha_{u,v}, u) \;\middle|\; (u, v) \in E \right\} \cup \left\{ (\alpha_{u,v}, \alpha_{u,w}) \;\middle|\; (u, v) \neq (u, w) \in E \right\}. \end{aligned}$$

Two observations are made about this construction. First, it has the property that $d'(u, v) = d(u, v) + 1$ holds for any pair of vertices $u, v \in V$, where d and d' are the two distance functions in G and G' respectively. This equality can be verified by contradiction. Next, if $(u, v) \in E$, then any two midvertices $\alpha_{u,x}$ and $\alpha_{v,y}$ have

$$\begin{aligned} d'(\alpha_{u,x}, \alpha_{v,y}) &\leq d'(\alpha_{u,x}, \alpha_{u,v}) + d'(\alpha_{u,v}, \alpha_{v,y}) \leq 2; \\ d'(\alpha_{u,x}, v) &= d'(\alpha_{u,x}, \alpha_{u,v}) + d'(\alpha_{u,v}, v) \leq 2; \\ d'(\alpha_{v,y}, u) &= d'(\alpha_{v,y}, \alpha_{u,v}) + d'(\alpha_{u,v}, u) \leq 2. \end{aligned}$$

Note strict '<'s appear in above three inequalities when $x = v$ or $y = u$, and in the first inequality when $x = y$. It is not difficult to see that the whole reduction can be done in time $O(|V|^3)$. Figure 10 illustrates an example of the reduction.

We claim that G has an independent set I of size at least l if and only if G' has a 3-independent set I' of the same size. Suppose I with $|I| \geq l$ is an independent set in G. Then I is also a 3-independent set in G'. This follows from our first observation. Conversely, suppose I' with $|I'| \geq l$ is a 3-independent set in G'. Then the set I, produced by replacing each midvertex $\alpha_{u,v} \in I'$ with either u or v, is an independent set in G. To see this, assume there exists two vertices $u, v \in I$ such that $d(u, v) = 1$. Thus $d'(u, v) = d(u, v) + 1 = 2 < 3$;



(a) (b)

Figure 10: An example of reduction from Independent Set to 3-IS. (a) An instance of Independent Set. (b) The constructed 3-IS instance: α vertices are added to increase the distances between the original vertices by exactly one.

so either u or v, or both, must have replaced some midvertices in I'. Let $s, t \in I'$ be the two vertices corresponding to u and v before the replacement respectively; that is, $s = u$ or $\alpha_{u,x}$ and $t = v$ or $\alpha_{v,y}$ for some $x, y \in V$. According to our second observation, we always have $d'(s,t) \leq 2$. Thus we have reached a contradiction, since $s, t \in I'$. That $|I'| = |I| \geq l$ is easy to verify in a similar way. □

Combining Lemmas 3 and 4, we have the NP-completeness of 0-2 D-Set; this immediately resolves the complexity of our remaining subproblem of D-Set:

Theorem 5 *D-Set remains NP-complete even if* $|S| \leq 2$ *for all* $S \in C$.

3.6 Experiments

For geometric preprocessing, we implemented the plane sweep algorithm by Nievergelt and Preparata [47]. We modified the original algorithm so that the containing polygons of each swept region are maintained and propagated along during the sweeping.[8] The greedy approximation algorithm for Set Covering was implemented with a linked list to attain the running time $O(\sum_{S \in C} |S|)$. All code was written in Common Lisp and was run on a Sparcstation IPX.

We discuss simulation results on random polygons. These simulations empirically study how the number of sampling points varies with the "density" of polygons in the plane. The results suggest that the point sampling approach is most effective at sensing polygonal objects that have highly overlapping poses. Experiments on a Zebra robot are underway and the results will be presented in the near future.

[8]This implementation has the same worst-case running time as a different version described in Section 3.1 which obtains the containment information by traversing the planar subdivision after the sweeping. But the implementation version is usually more efficient in practice.

3.6.1 Simulation Results

To generate random polygons, we precomputed an arrangement of a large number (such as 100) of random lines using Edelsbrunner and Guibas's topological sweeping algorithm [20]. A random polygon was extracted as the first "valid" cycle occurring in a random walk on this line arrangement, after being scaled to some random perimeter. By "valid" we mean that the number of vertices in the cycle was no less than some small random integer. This constraint was introduced merely to allow a proper distribution of polygons of various sizes, for otherwise triangles and quadrilaterals would be generated with high probabilities according to our observations. In a sample run, a group of 1000 polygons generated (by this method) from an arrangement of 100 random lines had sizes in the range 3–30, with mean 5.545 and standard deviation 3.225.

All random polygons (or all random poses of a single polygon) in a test were bounded by some square, so that the "density", i.e., the degree of overlap, of these polygons mainly depended on their number as well as on the ratio between their average area and the size of the bounding square. Since polygons were generated randomly, the average area could be viewed as approximately proportional to the square of the average perimeter. The configuration of each polygon, say P, was assumed to obey a "uniform" distribution inside the square. More specifically, the orientation of P was first randomly chosen from $[0, 2\pi)$; the position of P was then randomly chosen from a rectangle inside the square consisting of all feasible positions at that orientation.[9]

To be robust against sensor noise, the sampling point of every region in the region basis was selected as the center of a maximum inscribed circle in that region. In other words, this sampling point had the maximum distance to the polygon bounding that region. It is not difficult to see that such a point must occur at a vertex of the generalized Voronoi diagram inside the polygon, also called its *internal skeleton* or *medial axis function.*[10] Also for sensing robustness, regions with area less than some threshold were not considered at the stage of region basis computation.[11] Though this thresholding traded off the completeness of sampling, it almost never resulted in the failure of finding a region basis once the threshold was properly set.

The first two groups of six tests gave a sense of the number of sampling points required when polygons are sparsely distributed in the plane. The results are summarized in Table 4. Every test in group (a) was conducted on distinct, i.e., non-congruent, random polygons with perimeters between $\frac{1}{4}$ and $\frac{3}{4}$ of the width of the bounding square; every test in group (b) was conducted on distinct poses of a single polygon with perimeter equal to $\frac{5}{8}$ of the width of the square. The scene of the last test from each group is displayed in Figure 11.

Without any surprise, the number of sampling points found were *around half* of the

[9]If the diameter of P is greater than the width of the square, then not every orientation is necessarily feasible. However, this situation was avoided in our simulations.

[10]The construction of the internal skeleton of a polygon is a special case of the construction of the generalized Voronoi diagram for a set of line segments, for which $O(n \log n)$ algorithms are given in [24], [38], and [56]. Since the maximum region size for a region basis turned out to be very small in the simulations, we only implemented an $O(n^4)$ brute force algorithm.

[11]We thresholded on the region area rather than the radius of a maximum inscribed circle merely to avoid the inefficient computation on the latter for all the regions in the planar subdivision.

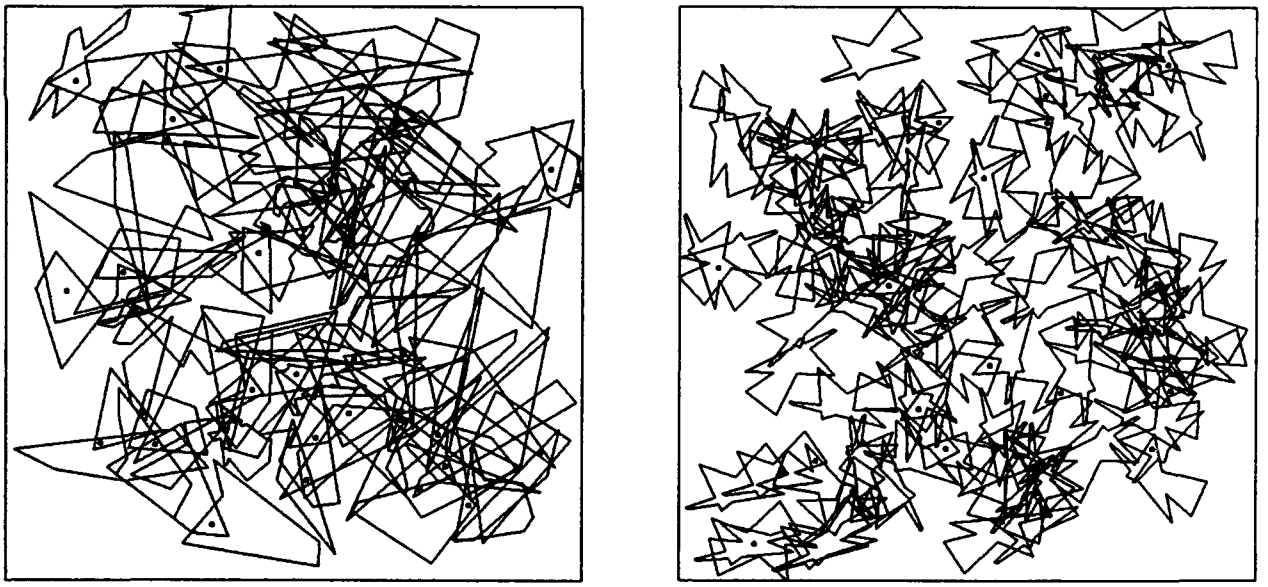
# polys	# regions	# sampling points
50	320	31
60	500	37
70	594	41
80	783	46
90	973	47
100	1422	51

(a)

# polys	# regions	# sampling points
50	362	36
60	609	34
70	741	34
80	1061	39
90	1125	49
100	1643	61

(b)

Table 4: Tests on sampling sparsely distributed random polygons/poses. The twelve tests were divided into two groups: (a) All polygons in each test were distinct, with perimeters between $\frac{1}{4}$ and $\frac{3}{4}$ times the width of the bounding square. (b) All polygons in each test represented distinct random poses of a same polygon. The polygon perimeter was uniformly $\frac{5}{8}$ times the side length of the square for all six tests in the group.



(a) (b)

Figure 11: Sampling 100 sparsely distributed random polygons/poses. (a) The scene of the last test from group (a) in Table 4: There are 1422 regions in the planar subdivision and 51 sampling points (drawn as dots) to discriminate the 100 polygons. (b) The scene of the last test from group (b) in Table 4: There are 1643 regions in the planar subdivision and 61 sampling points to discriminate the 100 poses.

number of polygons, for all twelve tests in Table 4. This supports the fact that, for n sparsely distributed polygons in the plane, the minimum number of sampling points turns out to be $\Theta(n)$. As we can see from Figure 11, in such a situation every polygon intersects at most a few, or more precisely, a constant number of, other polygons. In other words, the number of polygon pairs distinguishable by any single region in the planar subdivision is $\Theta(n)$; but there are $\lfloor \frac{n^2}{4} \rfloor$ such pairs in total! Thus sensing by point sampling is inefficient in a situation with a large number of sparsely distributed polygons.

The next two groups of six tests were on polygons much more densely distributed in the plane, and the results are given in Table 5. In these two groups of tests, we used a bounding

# polys	# regions	# sampling points
25	776	8
30	998	8
35	1270	12
40	1759	12
45	2153	11
50	2678	13

(c)

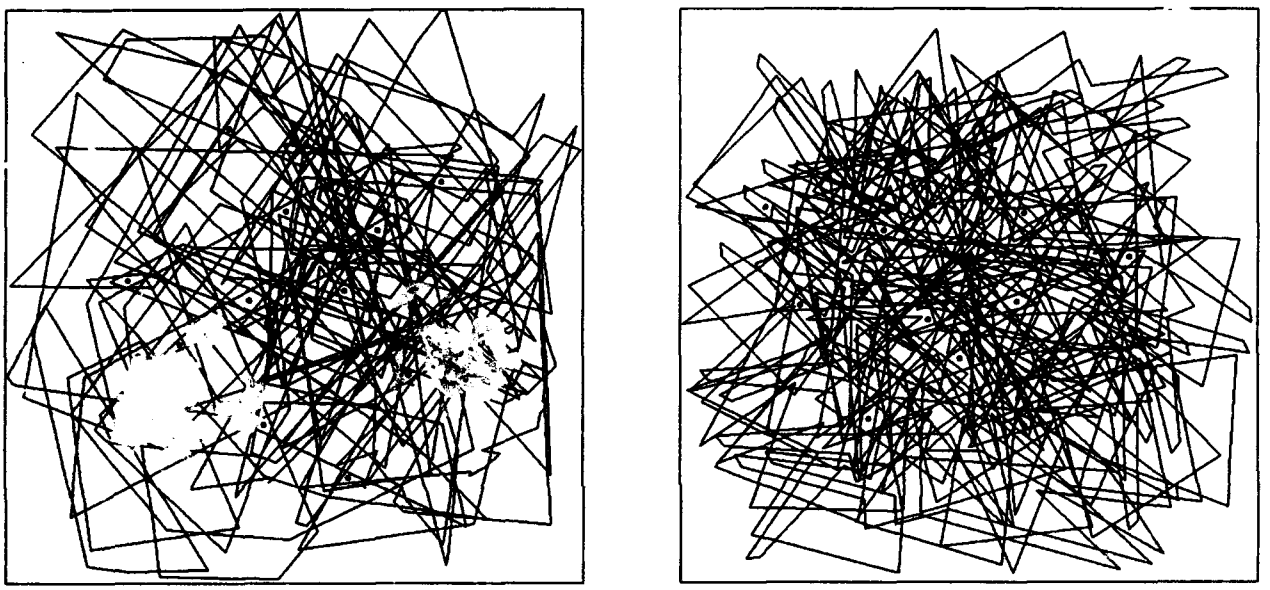
# polys	# regions	# sampling points
10	264	4
20	1121	7
25	1781	6
30	2796	7
35	3655	9
40	4995	9

(d)

Table 5: Tests on sampling densely distributed random polygons/poses, divided into two groups (c) and (d). The width of the bounding square was reduced to $\frac{1}{4}$ times the width of the square used in groups (a) and (b) in Table 4. In group (c) all polygons in each test were distinct with perimeters in the range between $\frac{1}{2}$ and 2 times the width of the bounding square. In group (d) all polygons in each test were distinct poses of the same polygon as in Figure 11(b).

square with side length only $\frac{1}{4}$ of the width of the one used in test groups (a) and (b). Every test in group (c) was conducted on distinct polygons with perimeters in the range $\frac{1}{2}$–2 times the side length of the bounding square. All tests in group (d) were distinct poses of the same polygon used in the last test of group (b). Again the scene of the last test from each group is shown in Figure 12.

All twelve tests in groups (c) and (d) except the last one in group (c) found sampling points *at most twice* the lower bound $\lceil \log n \rceil$, while the first test in group (d) found exactly $\lceil \log n \rceil$ sampling points. The data in group (d) were more densely distributed than the data in group (c) in that every two poses intersected each other. Since an extremely dense distribution of polygons may cause numerical instabilities in the plane sweep algorithm, smaller numbers of polygons were tested in these two groups than were tested in groups (a) and (b). The results of these two groups of tests show that the sampling strategy is very applicable to sensing densely distributed polygons.



(c) (d)

Figure 12: Sampling densely distributed polygons. The bounding square has width $\frac{1}{4}$ times the width of the one shown in Figure 11. (c) The scene of the last test from group (c) in Table 5: There are 50 distinct polygons which form a planar subdivision with 2678 regions, and which can be discriminated by 13 sampling points. (d) The scene of the last test from group (d) in Table 5: There are 40 distinct poses of the polygon from Figure 11(b), which form a planar subdivision with 4955 regions, and which can be discriminated by 9 sampling points.

4 Conclusion

The analyses and experiments in this paper have laid out the bases for two general sensing schemes applicable to planar objects with known shapes. The first scheme, termed *sensing by inscription*, determines the pose of an object by finding its inscription in a polygon of geometric constraints derived from the sensory data. An implementation may use a rotary sensor or a linear CCD array combined with a diverging lens to obtain the necessary constraints. In particular, two supporting cones are often enough to detect the real pose of a polygonal object. In real situations, if two (or more) possible poses arise from a two-cone inscription, they can be distinguished by point sampling.[12]

Though only the inscription of a convex polygon is treated the extensions to any arbitrary polygon and any polyhedron should be straightforward. In the second case the 3-D cones are defined by visible vertices as well as edges. But the extension to a closed and piecewise smooth curve needs further study. The technique can also be applied in object recognition: A finite set of polygons is generally distinguishable by inscription.

Robustness of inscription can be realized by allowing some tolerance for intersecting locus curves. That is, an intersection of two locus curves is considered to be a real pose if it is attained on these curves at orientations which differ by an amount less than the tolerance.

Future work will involve the design of specialized cone sensors or other sensors suited for inscription, as well as an investigation of a theoretical framework for incorporating sensing uncertainties into the inscription algorithms.

The second scheme, termed *sensing by point sampling*, distinguishes a finite set of poses by examining the containment of a number of points. It can be generalized to work for objects with more general boundaries as long as the planar subdivision can be efficiently constructed. Robustness mainly depends on the invariance of the topology of the planar subdivision. Therefore it is often better to consider only regions whose maximum inscribed circles are larger than a certain threshold.

Future work will combine existing mechanical orienting methods with point light detectors or mechanical probes for pose detection.

Acknowledgments

Section 2 is based on our paper [33] in the 1994 IEEE International Conference on Robotics and Automation. We wish to thank Matt Mason for his insights and for helping us better interpret the experimental data, and Mel Siegel for suggesting a possible optical implementation using a linear CCD coupled with a diverging lens, and Steve Shafer for offering useful feedback on the presentation of [33].

Many thanks to David S. Johnson for pointing to the results of [44] on the hardness of approximating Set Covering and for looking into the status of Discriminating Set and k-Independent Sets, which we found neither in the catalog [25] nor in the NP-Completeness

[12]Notice that cases with more than two possible poses never occurred in our experiments on two-cone inscription.

Columns of *Journal of Algorithms* starting from [35]. Also thanks to Somesh Jha for his valuable suggestions on the proof of Lemma 4, and to Bruce Donald for his valuable comments and reading of [32], on which Section 3 is based.

After presentation of [32] at the First Workshop on Algorithmic Foundations of Robotics we learned that similar results to our Theorem 3 and our approximation algorithm had appeared in [8], [50], [1], and [3]. We are very grateful to Jean-Daniel Boissonnat, Mark Overmars, Anil Rao, and Kathleen Romanik for pointing out these papers. We are particularly grateful to Kathleen Romanik for numerous interesting discussions on shape discrimination and shattered sets, and for her reading of [32].

References

[1] Esther M. Arkin, Patrice Belleville, Joseph S. B. Mitchell, David Mount, Kathleen Romanik, Steven Saltzberg, and Diane Souvaine. Testing simple polygons. In *Proceedings of the Fifth Canadian Conference on Computational Geometry*, 1993.

[2] Esther M. Arkin, Paul Chew, Daniel P. Huttenlocher, Klara Kedem, and Joseph S.B. Mitchell. An efficiently computable metric for comparing polygonal shapes. *IEEE Journal of Robotics and Automation*, 13(3), 1991.

[3] Esther M. Arkin, Hank Meijer, Joseph S. B. Mitchell, David Rappaport, and Steven S. Skiena. Decision trees for geometric models. In *Proceedings of the Ninth Annual Symposium on Computational Geometry*, pages 369–378. ACM Press, 1993.

[4] S. Arora, C. Lund, R. Motwani, M. Sudan, and M.Szegedy. Proof verfication and intractability of approximation problems. In *Proceedings of the 33rd IEEE Symposium on Foundations of Computer Science*, pages 14–23, 1992.

[5] Francis Avnaim and Jean Daniel Boissonnat. Polygon placement under translation and rotation. Technical Report 889, INRIA Sophia-Antipolis, 1988.

[6] B.S. Baker, S.J. Fortune, and S.R. Mahaney. Polygon containment under translation. *Journal of Algorithms*, 7(4):532–548, 1986.

[7] M. Bellare, S. Goldwasser, G. Lund, and A. Russell. Efficient probabilistically checkable proofs and applications to approximation. In *Proceedings of the 25th Annual ACM Symposium on Theory of Computing*, pages 294–304, 1993.

[8] Patrice Belleville and Thomas C. Shermer. Probing polygons minimally is hard. *Computational Geometry: Theory and Applications*, 2(5):255–265, 1993.

[9] J.D. Boissonnat and M. Yvinec. Probing a scene of nonconvex polyhedra. *Algorithmica*, 8(3):321–342, 1992.

[10] R.C. Brost. Automatic grasp planning in the presence of uncertainty. *International Journal of Robotics Research*, 7(1):3–17, 1988.

[11] John F. Canny and Kenneth Y. Goldberg. A "RISC" paradigm for industrial robotics. Technical Report ESRC 93-4, University of California at Berkeley, 1993.

[12] Bernard Chazelle. The polygon containment problem. In Franco P. Preparata, editor, *Advances in Computing Research*, pages 1–32. Jai Press Inc., 1983.

[13] Bernard Chazelle and Herbert Edelsbrunner. An optimal algorithm for intersecting line segments in the plane. *Journal of the ACM*, 39(1):1–54, 1992.

[14] Homer H. Chen. Pose determination from line-to-plane correspondences: existence condition and closed-form solutions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):530–541, 1991.

[15] V. Chvátal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979.

[16] Richard Cole and Chee K. Yap. Shape from probing. *Journal of Algorithms*, 8(1):19–38, 1987.

[17] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms.* McGraw-Hill, 1990.

[18] D. Dobkin, H. Edelsbrunner, and C.K. Yap. Probing convex polytopes. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing*, pages 424–432. ACM Press, 1986.

[19] Bruce R. Donald, Jim Jennings, and Danilela Rus. Towards a theory of information invariants for cooperating autonomous mobile robots. In *Sixth International Symposium on Robotics Research*, October 1–5 1993. Hidden Valley, Pennsylvania.

[20] Herbert Edelsbrunner and Leonidas J. Guibas. Topologically sweeping an arrangement. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing*, pages 389–403. ACM Press, 1986.

[21] Michael Erdmann. Understanding action and sensing by designing action-based sensors. In *Sixth International Symposium on Robotics Research*, October 1–5 1993. Hidden Valley, Pennsylvania.

[22] Michael Erdmann and Matt Mason. An exploration of sensorless manipulation. *IEEE Journal of Robotics and Automation*, 4(4):369–379, 1988.

[23] Steven Fortune. Fast algorithms for polygon containment. In *Automata, Languages and Programming*, pages 189–198. Springer-Verlag, 1986. LNCS 194.

[24] Steven Fortune. A sweepline algorithm for Voronoi diagrams. *Algorithmica*, 2:153–174, 1987.

[25] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H. Free and Company, 1979.

[26] Peter C. Gaston and Tomás Lozano-Pérez. Tactile recognition and localization using object models: The case of polyhedra on a plane. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(3):257–266, 1984.

[27] Kenneth Y. Goldberg and Matthew T. Mason. Bayesian grasping. In *Proceedings of the 1990 IEEE International Conference on Robotics and Automation*, pages 1264–1269, 1990.

[28] Rajeev Govindan and Anil S. Rao. Optimal strategies for recognizing polygonal parts. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 3564–3569, 1994.

[29] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science.* Addison-Wesley, 1989.

[30] W.E.L. Grimson. *Object Recognition by Computer: the Role of Geometric Constraints.* MIT Press, 1990.

[31] David D. Grossman and Michael W. Blasgen. Orienting mechanical parts by computer-controlled manipulator. *IEEE Transaction on Systems, Man, and Cybernetics*, 5:561–565, 1975.

[32] Yan-Bin Jia and Michael Erdmann. The complexity of sensing by point sampling. In *The First Workshop on the Algorithmic Foundations of Robotics.* A. K. Peters, Boston, MA, 1994.

[33] Yan-Bin Jia and Michael Erdmann. Sensing polygon poses by inscription. In *Proceedings of 1994 IEEE International Conference on Robotics and Automation*, pages 1642–1649. IEEE Computer Society Press, 1994.

[34] David S. Johnson. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences*, 9:256–278, 1974.

[35] David S. Johnson. The NP-completeness column: an ongoing guide. *Journal of Algorithms*, 4:393–405, 1981.

[36] Richard M. Karp. Reducibility among combinatorial problems. In Raymond E. Miller and James W. Thatcher, editors, *Complexity of Computer Computations.* Plenum Press, 1972.

[37] Keiichi Kemmotsu and Takeo Kanade. Uncertainty in object pose determination with three light-stripe range measurements. Technical Report CMU-CS-93-100, CMU School of Computer Science, 1993.

[38] David G. Kirkpatrick. Efficient computation of continuous skeletons. In *Proceedings of the Twentieth Annual Symposium on Foundations of Computer Science*, pages 18–27, 1979.

[39] Phokion G. Kolaitis and Madhukar N. Thakur. Approximation properties of NP minimization classes. In *Proceedings of the 6th Conference on Structure in Complexity Theory*, pages 353–366, 1991.

[40] S.-Y. R. Li. Reconstruction of polygons from projections. *Information Processing Letters*, 28:235–240, 1988.

[41] Michael Lindenbaum and Alfred Bruckstein. Parallel strategies for geometric probing. *Journal of Algorithms*, 13(2):320–349, 1992.

[42] Seppo Linnainmaa, David Harwood, and Larry S. Davis. Pose determination of a three-dimensional object using triangle pairs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):634–647, 1988.

[43] Lászo Lovász. On the ratio of optimal integral and fractional covers. *Discrete Mathematics*, 13:383–390, 1975.

[44] Carsten Lund and Mihalis Yannakakis. On the hardness of approximation minimization problems. In *Proceedings of the 25th Annual ACM Symposium on Theory of Computing*, pages 286–293. ACM Press, 1993.

[45] D. Mumford. The problem of robust shape descriptors. In *Proceedings of the First International Conference on Computer Vision*, pages 602–606, 1987.

[46] B.K. Natarajan. On detecting the orientation of polygons and polyhedra. In *Proceedings of the 3rd Annual Symposium on Computational Geometry*, pages 146–152, 1987.

[47] J. Nievergelt and F.P. Preparata. Plane-sweep algorithms for intersecting geometric figures. *Communications of the ACM*, 25(10):739–747, 1982.

[48] Franco P. Preparata and Michael Ian Shamos. *Computational Geometry: an Introduction.* Springer-Verlag, 1988.

[49] Anil S. Rao and Kenneth Y. Goldberg. Placing registration marks. In *Proceedings of 1993 IEEE International Conference on Robotics and Automation*, pages 161–167, 1993.

[50] Kathleen Romanik. Approximate testing theory. Technical Report CS-TR-2947/UMIACS-TR-92-89, University of Maryland at College Park, August 1992.

[51] Kathleen Romanik and Steven Salzberg. Testing orthogonal shapes. In *Proceedings of the Fourth Canadian Conference on Computational Geomery*, pages 216–222, 1992.

[52] David M. Siegel. Finding the pose of an object in a hand. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pages 406–411, 1991.

[53] F.W. Sinden. Shape information from rotated scans. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(6):726–730, 1985.

[54] Steven S. Skiena. Problems in geometric probing. *Algorithmica*, 4(4):599–605, 1989.

[55] Aaron S. Wallack, John F. Canny, and Dinesh Manocha. Object localization using cross-beam sensing. In *Proceedings of the 1993 IEEE International Conference on Robotics and Automation*, pages 692–699, 1993.

[56] Chee K. Yap. An $O(n \log n)$ algorithm for the Voronoi diagram of the set of simple curve segments. *Discrete & Computational Geometray*, 2:365–393, 1987.

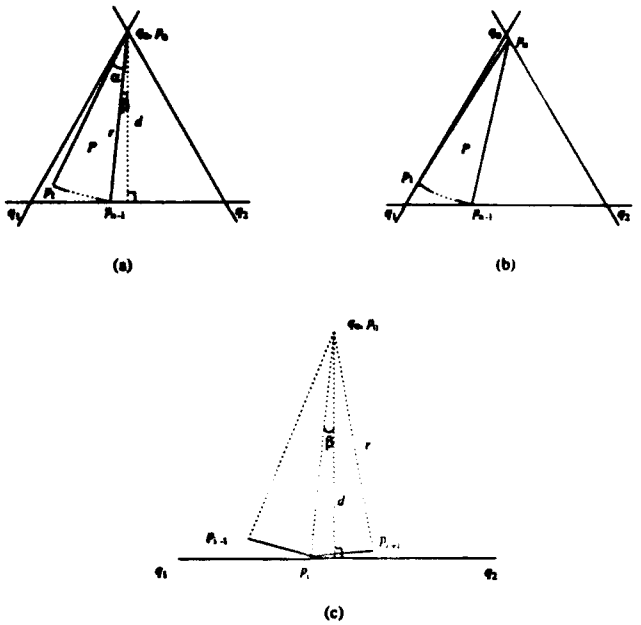


Figure 13: A worst case example of inscription. The three half-planes form an equilateral triangle $\triangle q_0 q_1 q_2$ by intersection and the convex n-gon P has vertex p_0 at the center of an arc equally divided by the remaining vertices $p_1, \ldots, p_{n-1}$ of P. The conditions derived in this appendix on radius r and measure α of the arc guarantee $6n$ different poses in which P is inscribed in $\triangle q_0 q_1 q_2$: (a) $6(n-1)$ poses of P result from rotations about p_0 positioned at vertices q_0, q_1, and q_2 respectively; and (b) 6 other poses result when p_0, p_1 and p_{n-1} are incident on different edges of $\triangle q_0 q_1 q_2$. (c) illustrates a pose of the first type such that vertex p_i, $2 \leq i \leq n-2$, is on edge $q_1 q_2$ during the rotation about q_0.

Appendices

A A Worst Case of Inscription

In this appendix we construct an example where a convex n-gon embedded in three half-planes can attain $6n$ possible poses.

Symmetry turns out to play a central role in the construction. Let the three half-planes form an equilateral triangle $\triangle q_0 q_1 q_2$ by intersection; and let P be a convex polygon with vertices $p_0, p_1, \ldots, p_{n-1}$ in counterclockwise order such that p_0 is the center of an arc of radius r and measure α, and $p_1, \ldots, p_{n-1}$ together divide this arc into $n-2$ equal pieces. The reason for choosing this particular shape of P is that we expect to obtain $2(n-1)$ poses by positioning p_0 at each of q_0, q_1, and q_2 and rotating P about p_0 inside $\triangle q_0 q_1 q_2$ such that $p_1, \ldots, p_{n-1}$ will all touch the opposite edge of $\triangle q_0 q_1 q_2$ exactly twice during the rotation. Figure 13(a) illustrates the first of the sequence of $2(n-1)$ poses resulting from the rotation around q_0. By symmetry we already have $6(n-1)$ poses in total, called poses of the *first* type. The remaining 6 poses of the *second* type are symmetric to each other, attained when vertices p_0, p_1 and p_{n-1} are on different edges of $\triangle q_0 q_1 q_2$. (See Figure 13(b).) So our task is to derive the conditions on radius r and measure α of the arc that realize these two types

of poses.

Let us first look at poses of the first type. Let $d > 0$ be the altitude of $\triangle q_0q_1q_2$ and β an acute angle with $\cos\beta = \frac{d}{r}$, and suppose p_0 coincides with q_0. We require $d < r$ and $\alpha < \alpha + \beta < \frac{\pi}{6}$ so that p_1 in the initial pose is not on edge q_0q_1 in order to allow poses of the second type shown in Figure 13(b), and p_{n-1} lies to the left of the midpoint on edge q_1q_2 in order to be incident on this edge the second time as P rotates counterclockwise. We thus have the following conditions on α and r:

$$0 < \alpha < \frac{\pi}{6} \qquad \text{and} \qquad \cos(\frac{\pi}{6} - \alpha) < \frac{d}{r} < 1.$$

With the above constraints, we still need to make sure that P is indeed above edge q_1q_2 when each vertex p_i, $1 \leq i \leq n-1$, becomes incident on this edge during the rotation. In fact, it suffices to make sure that vertices p_{i-1} and p_{i+1} are above edge q_1q_2. Figure 13(c) illustrates the case in which p_i lies to the left of the midpoint on edge q_1q_2. It is clear that p_{i-1} is above edge q_1q_2; p_{i+1} is above edge q_1q_2 if and only if $2\beta < \angle p_iq_0p_{i+1} = \frac{\alpha}{n-2}$, and thus if and only if

$$\frac{d}{r} > \cos\frac{\alpha}{2(n-2)}.$$

Starting from the initial pose shown in Figure 13(a), slide p_0 down along edge q_0q_2 and p_{n-1} left along edge q_1q_2 until p_1 touches edge q_0q_1; then we get the pose in Figure 13(b). Observe that p_0 is on edge q_0q_2 but closer to q_0 than to q_2, thereby ensuring the other 5 poses of the second type by symmetry.

Hence P has $6n$ feasible poses when inscribed in $\triangle q_0q_1q_2$ if the following two conditions are satisfied:

$$0 < \alpha < \frac{\pi}{6};$$
$$d < r < \min\Big(\frac{d}{\cos(\frac{\pi}{6} - \alpha)}, \frac{d}{\cos\frac{\alpha}{2(n-2)}}\Big).$$

In fact, the above conditions are also necessary for the given shapes of P and $\triangle q_0q_1q_2$.

Using the above idea, we can similarly construct a worst case example for the special case of inscription in which two of the three bounding lines are parallel; it was stated in Section 2.2.3 that there exist up to $4n$ poses for a convex n-gon. Let the three bounding lines be l_1, l_2 and l_3 such that $l_1 \parallel l_2$ and $l_3 \perp l_1, l_2$, and let a convex n-gon P take the same shape just described. Similarly we have that P has $4n$ feasible poses, $4(n-1)$ of which result from rotations about p_0 incident on l_1 and l_2 respectively, and 4 of which result when p_1 on one of l_1 and l_2 and p_{n-1} on the other, for any radius r and measure α that satisfy the following two conditions:

$$\xi < \alpha < \frac{\pi}{2};$$
$$\max\Big(d, \frac{d}{2\sin\frac{\alpha}{2}}\Big) < r < \min\Big(\frac{d}{\cos\frac{\alpha}{2(n-2)}}, \frac{d}{\sin\alpha}\Big),$$

where $\xi \in [0, \frac{\pi}{2})$ is the root of equation $2\sin\frac{\xi}{2} = \cos\frac{\xi}{2(n-2)}$, and d is the distance between l_1 and l_2.

B k-Independent Sets

We extend Lemma 4 to all k-IS with $k > 3$: They are NP-complete as well. The proof we will present is indeed a generalization of the proof of Lemma 4; it will again construct a k-IS instance with graph G' from an instance of Independent Set with graph G by local replacement. In the proof, each vertex v in G will be replaced by a simple path $\mathcal{P}_v$ of fixed length (depending only on k) that has v in the middle and an equal number of auxiliary vertices on each side; and each edge (u, v) will be replaced by four edges connecting the two end vertices on $\mathcal{P}_u$ with the two end vertices on $\mathcal{P}_v$, either directly or through a "midvertex". More intuitively speaking, all shortest paths between pairs of vertices in G, if they exist, get elongated in G' to such a degree that (1) (u, v) is an edge in G if and only if the distance between vertices u and v in G' is less than k; (2) any two vertices u' and v' in G' with a distance of at least k can be easily mapped to two *non*adjacent vertices in G. The first condition ensures that any given independent set in G will be a k-independent set in G', while the second condition ensures the construction of an independent set in G from any given k-independent set in G'.

Lemma 5 *k-Independent Set is NP-complete for all integers $k > 3$.*

Proof. Given an instance of Independent Set as a graph $G = (V, E)$ and a positive integer $l \leq |V|$, a k-IS instance is constructed by two consecutive substitutions. A path

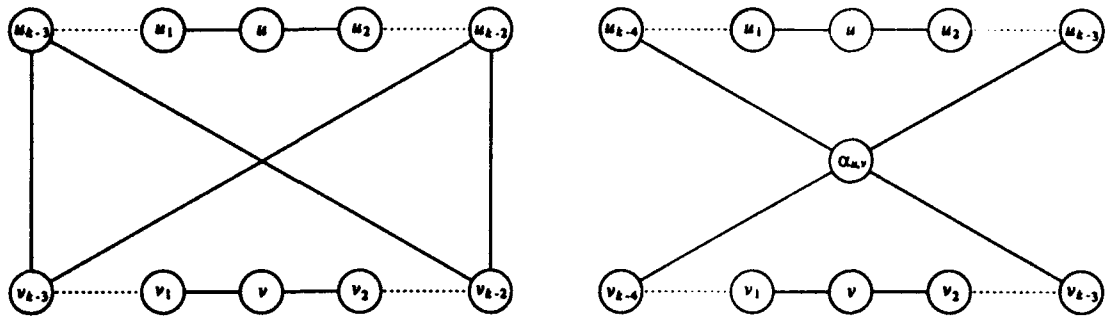
$$\mathcal{P}_v = \begin{cases} v_{k-3} \ldots v_1 v v_2 \ldots v_{k-2}, & \text{if } k \text{ even;} \\ v_{k-4} \ldots v_1 v v_2 \ldots v_{k-3}, & \text{if } k \text{ odd,} \end{cases}$$

first substitutes for vertex $v \in V$, where $v_1, \ldots, v_{k-3}$ (and v_{k-2} when k is even) are auxiliary vertices. And then a set of four edges

$$E_{u,v} = \begin{cases} \big\{(u_{k-3}, v_{k-3}), (u_{k-3}, v_{k-2}), \\ \quad (u_{k-2}, v_{k-3}), (u_{k-2}, v_{k-2})\big\}, & \text{if } k \text{ even;} \\ \big\{(u_{k-4}, \alpha_{u,v}), (u_{k-3}, \alpha_{u,v}), \\ \quad (v_{k-4}, \alpha_{u,v}), (v_{k-3}, \alpha_{u,v})\big\}, & \text{if } k \text{ odd,} \end{cases}$$

substitute for each edge $(u, v) \in E$, where $\alpha_{u,v}$ is an introduced midvertex. Figure 14 shows two subgraphs after applying the above substitutions on edge $(u, v) \in E$, for k even and odd respectively.

We can easily verify that, for any pair of vertices x on $\mathcal{P}_u$ and y on $\mathcal{P}_v$, both k even and odd, we have $d'(x, y) \leq d'(u, v) = k - 1 < k$ if $(u, v) \in E$, where d' is the distance function defined on G'. On the other hand, if $(u, v) \notin E$, we have $d'(u, v) \geq k$ when k is even and $d'(u, v) \geq k+1$ when k is odd. Thus an independent set I in G is also a k-independent set in G'. Conversely, suppose I' with $|I'| \geq l$ is a k-independent set in G. We substitute $u \in V$ for every auxiliary vertex $u_i \in I$ on path $\mathcal{P}_u$, and u or v for every midvertex $\alpha_{u,v} \in I$ when k is odd. Let I be the set after this substitution. It needs to be shown that I is an independent set in G and $|I| = |I'| \geq l$. This is obvious for the case that k is even. When k is odd,



(a) (b)

Figure 14: Two subgraphs resulting from the described substitutions performed on edge $(u, v) \in E$ for the cases that (a) k is even; and (b) k is odd.

however, the situation is a bit more complicated due to the possible occurrences of those α vertices in I'. We observe, for any $\alpha_{u,v}$, $\alpha_{u',v'}$, and x on path $\mathcal{P}_w$ where $u, v, u', v', w \in V$,

$$\begin{aligned} d'(\alpha_{u,v}, x) &\leq \frac{k-3}{2} + 3 < k, && \text{if } w = u \text{ or } v, \text{ or } (u,w) \in E, \text{ or } (v,w) \in E; \\ d'(\alpha_{u,v}, \alpha_{u',v'}) &\leq 4 < k, && \text{if } (u,u') \in E. \end{aligned}$$

In fact, these two conditions guarantee that I is an independent set in G and $|I| = |I'|$, which we leave for the reader to verify.

The reduction can be done in time $O(k|V| + |E|)$, which reduces to $O(|V| + |E|)$ if k is treated as a constant, in contrast to the time $O(|V|^3)$ required for the reduction from Independent Set to 3-IS. This time reduction is due to the fact that midvertices corresponding to the same vertex in V no longer have edges between each other. □

Since 1-IS can be easily solved by comparing $|V|$ and l, we are now ready to sum up the complexity results on this family of problems in the following theorem.

Theorem 6 *k-Independent Set is in P if $k = 1$ and NP-complete for all $k \geq 2$.*